NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

DESIGN OF A SYNCHRONOUS PIPELINED MULTIPLIER
AND ANALYSIS OF CLOCK SKEW IN HIGH-SPEED
DIGITAL SYSTEMS

by

John R. Calvert, Jr.

December 2000

Thesis Advisor: Douglas J. Fouts
Thesis Co-Advisor: Herschel H. Loomis, Jr.

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December 2000	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE : Design of a Synchronous Pipelined Multiplier and Analysis of Clock Skew in High-Speed Digital Systems	5. FUNDING NUMBERS
6. AUTHOR(S) Calvert, John R. Jr.	

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING /MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.	12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

Digital systems implemented with high-speed transistor technologies face a variety of design challenges in an effort to keep pace with the accelerating demand for performance. As device switching frequencies climb comfortably into the gigahertz range, clock skew in digital systems threatens to limit the advantages of synchronous pipelined designs. This research investigates the limitations of clock skew on high-speed digital systems by designing and simulating an 8x8 bit synchronous, pipelined multiplier using Indium phosphide (InP), heterostructure bipolar junction (HBT) transistor technology. Fundamentals of circuit analysis and the principles of junction transistor behavior are applied to design an optimal family of logic devices using current-mode logic. All testing and simulation data is based upon results obtained from Tanner SPICE design tools. Using the building blocks of this logic family, an array multiplier is constructed and further configured into five distinct pipeline implementations. By employing a different number of pipeline stages in each implementation, the trade-offs of pipelining are illustrated and clock skew is analyzed at a variety of throughput rates. Finally, the impact of clock skew on throughput performance is quantified and summarized as a reference point for further research into asynchronous control techniques.

14. SUBJECT TERMS Clock Skew, Pipelined Logic, Current-Mode Logic, Indium-phosphide Heterojunction Bipolar Transistors, High-Speed Logic	15. NUMBER OF PAGES 152
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

i

THIS PAGE LEFT BLANK INTENTIONALLY

DESIGN OF A SYNCHRONOUS PIPELINED MULTIPLIER AND ANALYSIS OF CLOCK SKEW IN HIGH-SPEED DIGITAL SYSTEMS

John R. Calvert, Jr.
Major, United States Marine Corps
B.S., United States Naval Academy, 1990

Submitted in partial fulfillment of the
requirements for the degree of

## MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
### December 2000
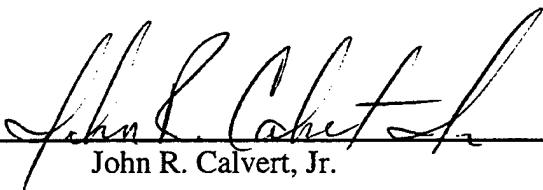
Author: _____
John R. Calvert, Jr.

Approved by: _____
Douglas J. Fouts, Thesis Advisor
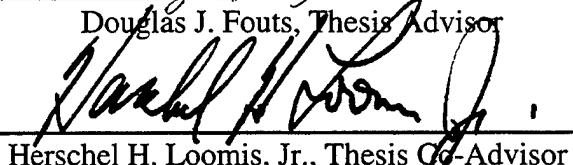
_____
Herschel H. Loomis, Jr., Thesis Co-Advisor
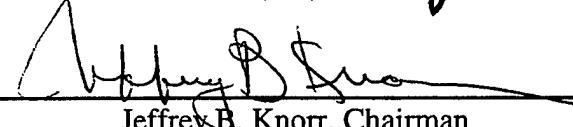
_____
Jeffrey B. Knorr, Chairman
Department of Electrical and Computer Engineering

iii

ABSTRACT

Digital systems implemented with high-speed transistor technologies face a variety of design challenges in an effort to keep pace with the accelerating demand for performance. As device switching frequencies climb comfortably into the gigahertz range, clock skew in digital systems threatens to limit the advantages of synchronous pipelined designs. This research investigates the limitations of clock skew on high-speed digital systems by designing and simulating an 8x8 bit synchronous, pipelined multiplier using Indium phosphide (InP), heterostructure bipolar junction (HBT) transistor technology. Fundamentals of circuit analysis and the principles of junction transistor behavior are applied to design an optimal family of logic devices using current-mode logic. All testing and simulation data is based upon results obtained from Tanner SPICE design tools. Using the building blocks of this logic family, an array multiplier is constructed and further configured into five distinct pipeline implementations. By employing a different number of pipeline stages in each implementation, the trade-offs of pipelining are illustrated and clock skew is analyzed at a variety of throughput rates. Finally, the impact of clock skew on throughput performance is quantified and summarized as a reference point for further research into asynchronous control techniques.

THIS PAGE LEFT BLANK INTENTIONALLY

TABLE OF CONTENTS

EXECUTIVE SUMMARY

The electronic subsystems of future overhead collection platforms will require extremely high performance digital logic for performing such tasks as data compression/decompression, data encryption, spread spectrum modulation, etc. To accomplish this, bit rates must reach into the gigabits per second range. Such speed obviously requires digital logic which will function correctly at clock rates of tens of gigahertz. The need for such high performance has led to the implementation of logic systems using indium phosphide (InP) heterojunction bipolar transistors (HBT) technology. However, clock frequency and pipeline throughput in digital systems implemented with InP HBT technology is significantly limited by clock, control signal, and data skew which is a much larger percentage of the clock period than it is in lower-speed digital systems implemented with complementary metal oxide semiconductor (CMOS) technology. Therefore, the presence of clock skew in high-speed digital systems defines a limitation for the advantages of synchronous pipelined architectures.

It is the purpose of this thesis to design a synchronous 8x8 bit pipelined multiplier as a high-speed digital test circuit using InP HBT technology and furthermore, to quantify the impact of clock skew on throughput. This work represents the initial phase of a larger research project to determine if asynchronous pipeline control will yield greater overall pipeline throughput in high-performance InP HBT digital integrated circuits and if the resulting elimination of the clock distribution tree will reduce power consumption, device count and layout area. All simulation data is based upon the results obtained from Tanner SPICE design tools.

Having received InP HBT device specifications from Hughes Research Laboratories, this project commenced with the design of an HBT logic family utilizing current-mode logic. Each circuit was designed and optimized for a minimum power-delay product while driving a maximal fanout load of four logic gates. This design effort produced the four essential circuit functions necessary for the practical implemention of any synchronous logic circuit: an inverter/buffer gate, an OR/NOR gate, a D-type latch, and a practical current source.

Using the building blocks of this logic family, an array multiplier was constructed and further configured into five distinct pipeline implementations. These included a one, two, four, six, and ten-stage pipeline, respectively. A comparative analysis of their performance effectively illustrated the trade-offs of pipelining, i.e., the cost of the additional registers was shown to outpace the increase in throughput beyond a six-stage implementation. At a maximum throughput of 4.35 gigahertz, the six-stage pipelined multiplier was the most efficient design (in the absence of clock skew). The highest throughput achieved was 5.56 gigahertz by the costly ten-stage implementation. Power consumption ranged from 4.4 to 14 watts.

In the final analysis, clock skew was not simulated because SPICE simulations effectively eliminate skew from their calculations. Rather, the impact of clock skew was determined by applying numerical analysis to the no-skew simulation results. A range of possible skew values was considered in order to demonstrate a performance trend. The results confirmed that digital system throughput rates which are obtained as a function of higher clock rates will experience the most drastic performance reductions in the presence of clock skew. Also, it was shown for a typical

value of skew in this circuit that the efficiency curve shifts to indicate that the four-stage pipeline is the most efficient implementation, vice the six-stage pipeline.

The design products and test results from this thesis provide a reference point for further research into alternative clocking/control techniques. Specifically, it is intended that future research use the CML HBT logic family designed in this thesis in order to implement the same array multiplier circuit using asynchronous control techniques. One such endeavor is already in progress as LtCol. Kirk Shawhan, USMC, investigates the use of local completion signals which employ request/acknowledge handshake signals to control the flow of data vice the use of a global clock signal.

THIS PAGE LEFT BLANK INTENTIONALLY

ACKNOWLEDGMENT

My most immediate expression of gratitude is to the One that made me … to my Creator, my Lord, and my Savior — Jesus Christ. I am convinced that the meticulous detail and extensive effort required to design even the simplest of electronic devices bears unmistakable testimony to the Divine and Intelligent design of our universe.

But remember the Lord your God, for it is he who gives you the ability …   Deuteronomy 8:18

…whatever you do, do it all for the glory of God.
I Corinthians 10:31

This thesis is dedicated to my wife, Laura. Though I have no expectation that she will ever read beyond this page, certainly it would never have been written without her encouragement, support, and patience. While I have considered myself fortunate to be learning about the mysteries of microelectronics, she has been making the most thrilling discoveries and profound impact each day as the mother of our beloved daughter, Victoria, who is 22 months old today. I love them both dearly.

I wish to thank my advisors, Professor Douglas Fouts and Professor Herschel Loomis for sharing their time, their knowledge, and their passion for this field of study. Their guidance was essential and their enthusiasm was contagious.

I am also thankful to have shared this academic journey with such a quality core of fellow students and service members. Specifically, it is with great admiration that I thank LtCol. Kirk Shawhan, USMC — my perpetual study partner and project mentor. He has patiently endured my questions and generously shared his ingenius grasp of the most challenging concepts. My graduate school experience would have been a much different story without his assistance and more importantly, his friendship.

THIS PAGE LEFT BLANK INTENTIONALLY

I.   INTRODUCTION

A.   THE RELEVANCE OF HIGH-SPEED LOGIC

The demand for increased processing speeds in digital electronics has driven the clock frequency of logic circuits from a scale of microseconds to one of picoseconds over the past twenty years. This remarkable trend is the synergistic result of technological advancements and innovations in device physics, very-large-scale integrated (VLSI) circuit fabrication, and digital systems architecture. Moore's Law accurately predicted this trend of improvement 35 years ago, and current expectations are that the trend will continue (Moore, 1997). Consider the anticipation of such technologies as real-time multimedia satellite communications and broadband networks. These applications will require extremely high performance digital logic that can function reliably at clock rates of tens of gigahertz.

B.   THE PROBLEM OF CLOCK SKEW

There are a variety of technological hurdles to clear before achieving such clock speeds, and it is the purpose of this thesis to explore one particular hurdle in the course of digital systems architecture: the problem of clock skew in high-speed logic. Clock skew is the difference between arrival times of the clock signal at different synchronous clocked devices (Harris, 1999). As clock frequencies reach

1

into the multi-gigahertz range, clock skew is an increasing concern for high-speed circuit designers because it accounts for an increasing portion of the clock period — leaving less of the clock period to be budgeted for logic and latching delays. What was once a near negligible quantity has now become a significant design constraint. (Wakerly, 2000)

C.   THE DESIGN OF A TEST CIRCUIT

This thesis presents the design of a high-speed logic test circuit and the simulation of its performance in order to identify and quantify the effects of clock skew. It should be noted that these results are intended to serve as a reference for future research involving potential solutions for the reduction of clock skew. The following paragraphs develop the necessary specifications of the test circuit.

To ensure valid results, it is important that the problem be simulated in an accurate context. Therefore, it is necessary to select a logic family based upon a transistor model that is capable of realizing multi-gigahertz clock speeds. Although complementary metal-oxide-semiconductor (CMOS) technologies dominate VLSI applications, for comparable fabrication technologies, a bipolar circuit is approximately 2.5 times faster than a functionally similar CMOS circuit (Foley, 1994). Typically,

such high-speed bipolar circuits employ emitter coupled logic (ECL) or current mode logic (CML). Notably, these logic families consume significantly more power than field effect transistor (FET) logic families; however, the trade-off is accepted here for the purpose of achieving sufficient clock speeds. For these reasons, current mode logic is employed to design a family of logic gates based upon the transistor specifications for an indium phosphide (InP) heterojunction bipolar transistor (HBT), courtesy of Hughes Research Laboratories.

Additionally, it is important that the architecture and functionality of the test circuit provide a relevant context for evaluation. It should be noted here that the shorter clock periods discussed above are not exclusively the result of faster gate delays (i.e. faster transistors) but are also the result of pipelined architectures which require fewer gate delays per clock cycle. In keeping with this characteristic of high-speed logic circuits, the test circuit implements a pipelined architecture. As for circuit functionality, an 8x8 bit multiplier was chosen to provide sufficient complexity for pipeline implementation.

D.   THESIS OUTLINE

The purpose of this thesis is to design, simulate, and evaluate the performance of a high-speed (InP HBT) 8x8-bit pipelined multiplier in the presence of clock skew. The

3

discussion begins with the review and development of several fundamental topics in Chapter II: clock skew, pipelining principles, logic-level design of a multiplier, and transistor-level design of BJT/HBT logic. Based upon that foundation, Chapters III through V present the hierarchical design of the pipelined multiplier from the bottom up. Respectively, these chapters address logic circuit design, clock-driven circuit design, and pipeline design. Each of the design chapters presents a complete discussion of pertinent design issues, low-level simulation, performance optimization, and final design specifications. Finally, Chapter VI records the analysis of clock skew and Chapter VII summarizes the conclusions of the entire work.

II. BACKGROUND

A. CLOCK SKEW

Clock skew is the difference between the arrival times of the clock signal at two different clock-driven devices, as illustrated in Figure (2-1). This difference is dependent upon multiple issues including normal component variations, wire propagation delay, RC delays, propagation distance, environmental variations (such as operating temperature), and clock loading. Notably, all of these contributing factors have been increasing relative to gate delays. (Harris, 1999)
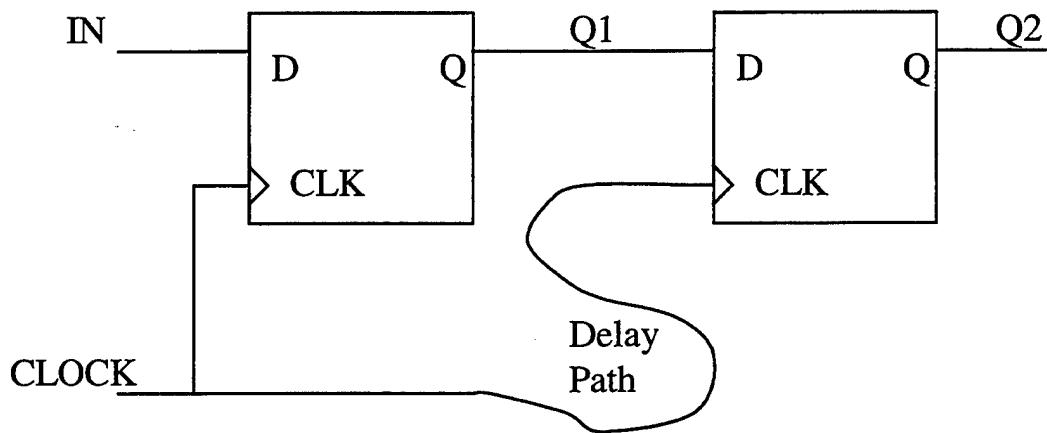
Figure 2-1. Clock Skew (After Wakerly).

In traditional logic designs which employ flip-flops and operate at extremely high clock frequencies, clock skew has become a significant portion of the total clock period.

For a fixed-length clock period, this effectively reduces the amount of time available for computation. Equation (2-1) quantifies the terms which contribute to the minimum clock period (T_{min}) of a traditional synchronous logic circuit.

(2-1)
$$T_{min} = t_{skew} + t_{logic} + t_{Flip-Flop}$$

$$\text{where, } t_{Flip-FLop} = t_{setup} + (t_{prop})_{max}$$

The simplest and most direct technique for minimizing clock skew would seem to be the implementation of a uniform clock distribution hierarchy which provides a local clock signal to a smaller portion of the entire circuit, i.e., a subcircuit. For signals that remain within the subcircuit, clock skew is reduced. The maximum propagation delay from the local clock source to the farthest clock input of the subcircuit can be kept within a desirable tolerance. But inevitably, signals must travel between subcircuits. This is an increasingly common occurrence when the maximum size of the subcircuit is restricted by practical limitations for fanout and power consumption — especially true in the case of current-driven logic.

The local clock signals are not without skew relative to each other. Although the delay paths for each branch of the clock distribution tree may contain the same number of gate delays, the switching behavior along each path varies

6

within a narrow range. Thus, when a signal from one subcircuit must drive logic in another subcircuit, the worst-case value of relative clock skew must be assumed.

An extensive clock distribution tree is employed in this thesis to provide local clock signals for circuit elements of a pipelined multiplier. Ultimately, the purpose is to quantify the clock skew experienced in a high-speed logic circuit and explore the impact of clock skew as the clock period is reduced.

B. PRINCIPLES OF PIPELINING

As referenced in the previous section, the minimum clock period is governed by the relationship presented in Equation (2-1). For a given block of combinational logic with an associated propagation time of t_{logic}, the minimum clock period is required to be even greater. In the face of a large, complex combinational circuit (Figure 2-2a) this could impose undesirable restrictions on clock speed.

However, a pipelined approach suggests that the combinational logic can be broken down into discrete levels of operation, known as pipeline levels (Figure 2-2b). Each pipeline level will contain fewer levels of logic than the original combinational circuit, and ideally, each pipeline level will contain the same number of logic levels in order to achieve near-equal propagation delays. Then, by adding appropriately sized registers between these levels (Figure

7

2-2c), the function of the original combinational logic can be achieved by sequentially sending operands through the series of pipeline levels.

Furthermore, this can be done at a higher clock rate since the period is now governed by Equation (2-2), where t_{logic} has now become $t_{pipe-level}$.

(2-2) $$T_{clock} = t_{skew} + t_{pipe-level} + t_{Flip-Flop}$$

The improvement in clock speed is quantified as the percentage of speedup, Equation (2-3). (Pollard, 1990)

(2-3)
$$Speedup = \frac{Time\ for\ M\ operations\ WITHOUT\ pipelining}{Time\ for\ M\ operations\ WITH\ pipelining}$$

Of course, this benefit is not without cost. There are several trade-offs involved such as increases in the number of components, power consumption, control complexity, chip area, and a variety of associated costs for design and fabrication. Additionally, the propagation latency for a single set of signals traveling through the pipeline is increased due to the additional delays contributed by the intermediate register(s) in the pipeline. Equation (2-4) expresses this increase in latency as a function of the number of pipeline stages (m) and the total register delay (Loomis, 2000).

(2-4) $$Latency\ Increase = (m-1)\, t_{Flip-Flop}$$

8
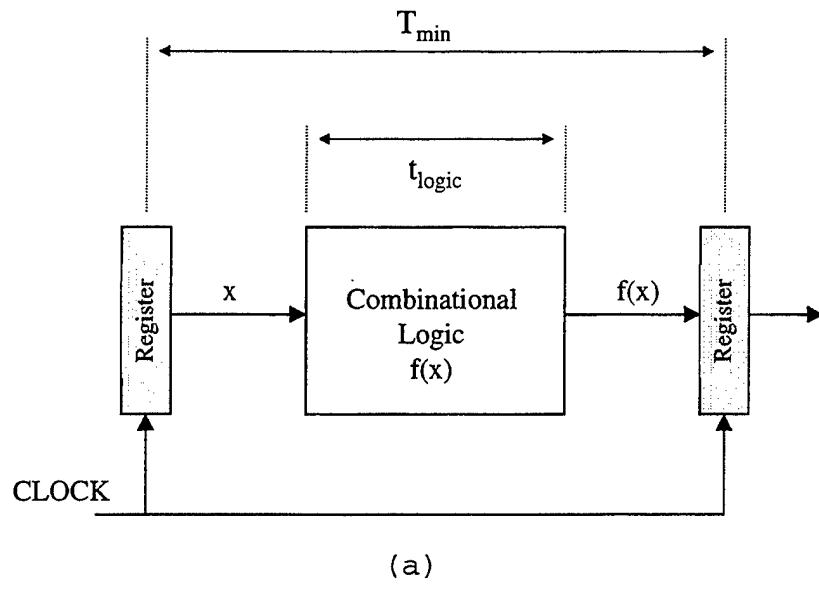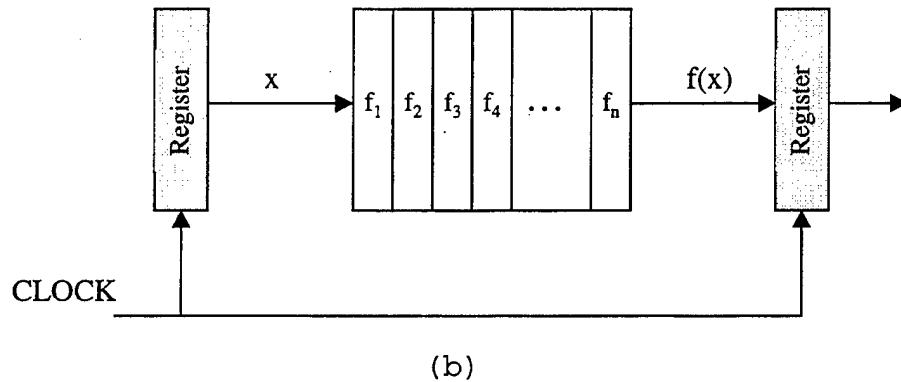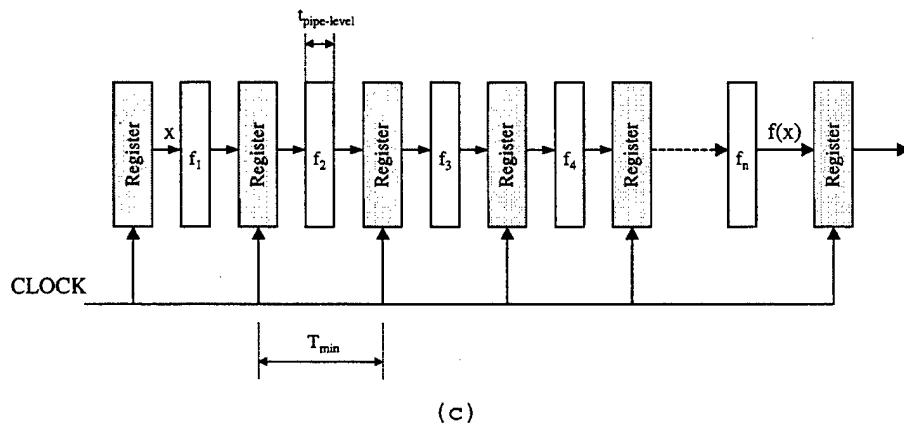
Figure 2-2. Example of Pipelining (After Loomis).

Though the significant increase in delay for a single operation may seem to be a tragic loss, it is the remarkable increase in data throughput which accompanies the increase in clock speed that ultimately motivates the designer to adopt a pipelined architecture.

In the context of this project, a pipelined architecture will facilitate the achievement of high clock speeds in the implementation of a relatively large, complex combinational circuit — a combinational multiplier.

C.   LOGIC DESIGN OF A COMBINATIONAL MULTIPLIER

A combinational multiplier takes two n-bit operands and performs n shift and n add operations to generate a $2n$-bit product. Most algorithms are implemented based upon the paper-and-pencil-like procedure of shifted product components as shown in Figure (2-3). Each individual bit of the multiplier (y_0 through y_{n-1}) is successively multiplied times the entire n-bit multiplicand. With each subsequent multiplier bit, the resulting product component is shifted by one bit position, starting with an initial shift of zero and concluding with $n-1$. (Wakerly, 2000)

The worst-case delay for this type of multiplication is governed by the carry propagation out of the most significant bit position and into the follow-on stage of addition. By utilizing carry-save addition (Figure 2-4), this propagation delay is eliminated for the initial $n-1$

$$
\begin{array}{cccccccccccccccc}
 & & & & & & & & y_0x_7 & y_0x_6 & y_0x_5 & y_0x_4 & y_0x_3 & y_0x_2 & y_0x_1 & y_0x_0 \\
 & & & & & & & y_1x_7 & y_1x_6 & y_1x_5 & y_1x_4 & y_1x_3 & y_1x_2 & y_1x_1 & y_1x_0 & \\
 & & & & & & y_2x_7 & y_2x_6 & y_2x_5 & y_2x_4 & y_2x_3 & y_2x_2 & y_2x_1 & y_2x_0 & & \\
 & & & & & y_3x_7 & y_3x_6 & y_3x_5 & y_3x_4 & y_3x_3 & y_3x_2 & y_3x_1 & y_3x_0 & & & \\
 & & & & y_4x_7 & y_4x_6 & y_4x_5 & y_4x_4 & y_4x_3 & y_4x_2 & y_4x_1 & y_4x_0 & & & & \\
 & & & y_5x_7 & y_5x_6 & y_5x_5 & y_5x_4 & y_5x_3 & y_5x_2 & y_5x_1 & y_5x_0 & & & & & \\
 & & y_6x_7 & y_6x_6 & y_6x_5 & y_6x_4 & y_6x_3 & y_6x_2 & y_6x_1 & y_6x_0 & & & & & & \\
+ & y_7x_7 & y_7x_6 & y_7x_5 & y_7x_4 & y_7x_3 & y_7x_2 & y_7x_1 & y_7x_0 & & & & & & & \\
\hline
p_{15} & p_{14} & p_{13} & p_{12} & p_{11} & p_{10} & p_9 & p_8 & p_7 & p_6 & p_5 & p_4 & p_3 & p_2 & p_1 & p_0
\end{array}
$$

Figure 2-3. Multiplication as a sum of partial product terms (From Wakerly).
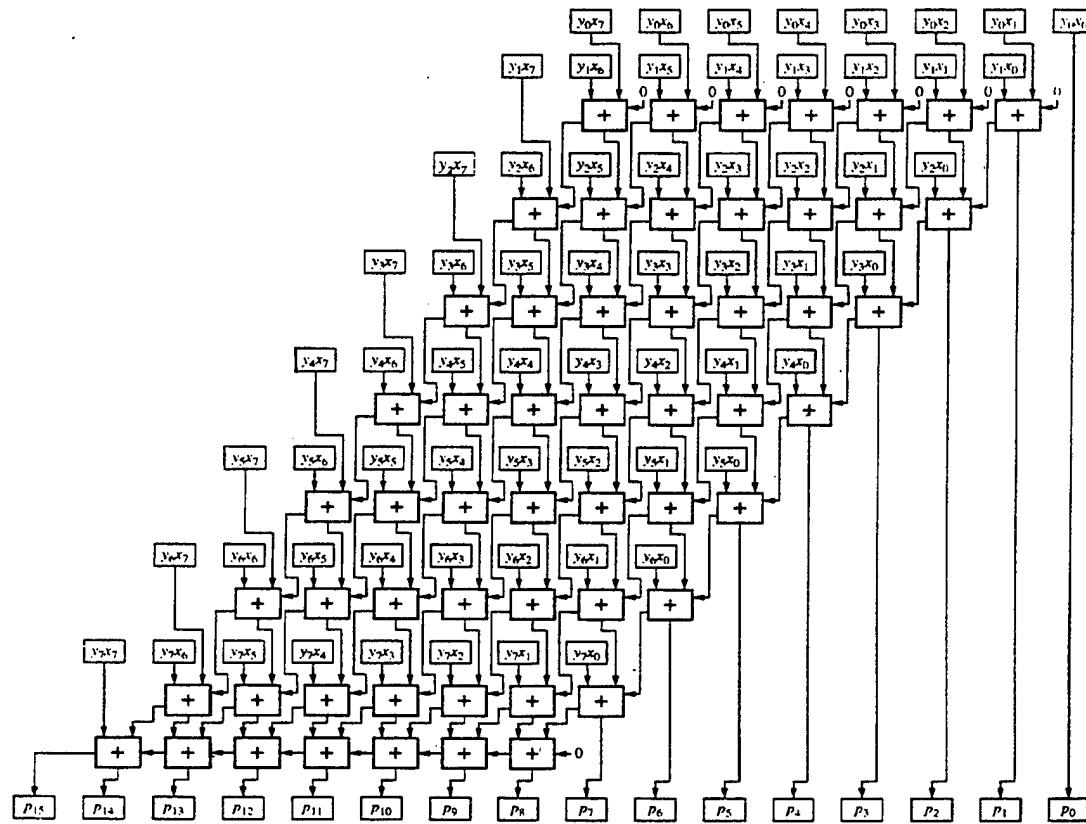
Figure 2-4. An 8x8 bit multiplier implemented with seven carry-save adder stages and one ripple-carry adder for carry completion (From Wakerly).

11

stages of addition; however, an extra stage is required to complete the addition of the final two resulting terms, as will be explained shortly.

The first carry-save addition stage takes two binary addends and generates an n-bit modulo-two sum and a shifted n-bit carry term (shifted by one bit). Subsequent carry-save addition stages take three binary addends: the previous partial sum, the shifted carry term, and the next subsequent product term. These are also added to produce an n-bit modulo-two sum and a shifted n-bit carry term. As each carry-save addition occurs, the least significant bit (LSB) of each partial sum represents the next most significant bit (MSB) in the final product. This is repeated until the nth product term has been added, and all that remains are a sum term and a shifted carry term. At this point, a carry-completion adder computes the most significant $n+1$ bits of the product. This procedure accounts for the consecutive propagation of a carry bit as each pair of addend bits are summed from LSB to MSB.

In the context of this project, the implementation of carry-save adders and carry completion adders allows convenient grouping of pipeline stages. This is particularly applicable to the final stage of the design process undertaken in this project. Chapter 5 provides further details on the implementation of a pipelined 8x8-bit

combinational multiplier, as introduced in the preceding paragraphs.

D.   BJT/HBT LOGIC

1.   BJT/HBT Principles and Characteristics

a)   *Device Structure*

A bipolar junction transistor (BJT) is a sandwich structure of three separately doped regions of silicon (or other suitable semiconductor), such that one of two configurations exists.   One configuration is the *pnp* transistor where a negatively doped region is bounded on either end by positively doped regions (p-type transistor). The other configuration is the *npn* transistor where a positively doped region is bounded on either end by negatively doped regions (n-type transistor).   Figure (2-5) provides a simplified illustration and further identifies the proper names for the regions: collector, base, and emitter.
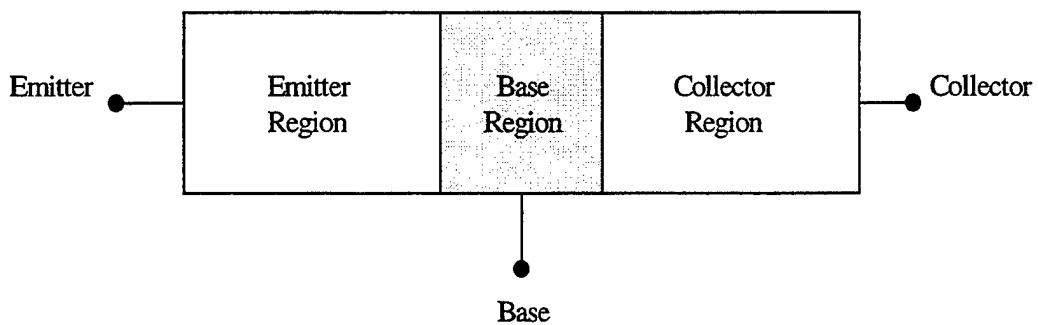
Figure 2-5.   Structure of a Bipolar Junction Transistor (After Pierret).

Until recent years, BJTs were generally fabricated from a single semiconductor material. However, device-level physics has demonstrated that faster junction transistors can be constructed from dissimilar semiconductor materials with complementary properties. Such devices are known as heterojunction bipolar transistors (HBTs). Conveniently enough, their operational behavior is essentially governed by the same functional principles as BJTs (Pierret, 1996). Therefore, it is assumed that wherever BJT behavior is referenced, a direct correspondence to HBT behavior exists. The following sections will provide a fundamental understanding of that behavior.

b) *Device Function*

The significance of the BJT lies in its potential to behave as a current-controlled current source when the proper DC bias is applied to the three regions or terminals. The controlling terminal is the base. Applying the proper DC bias to an *npn* transistor, a small current flowing into the base will produce a proportionately larger current being drawn into the collector, across the base region, and out of the emitter (Figure 2-6). The converse is true for a properly biased *pnp* transistor. A small current drawn out of the base will produce a proportionately larger current being drawn into the emitter, across the base region, and out of the collector. From this point forward, it will be helpful

14
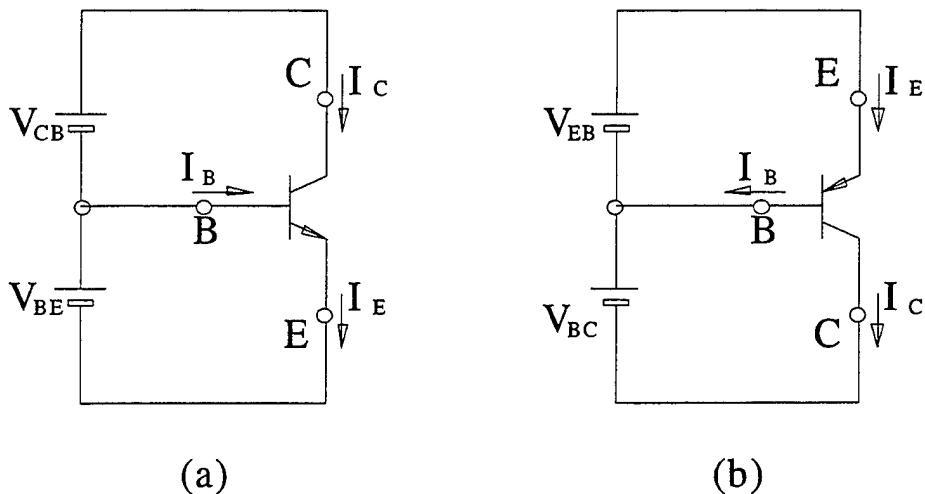
Figure 2-6. A functional illustration of an (a) npn and a (b) pnp bipolar junction transistor (After Sedra).

to limit the discussion to *npn* transistors, because the *pnp* transistors operate in a very similar manner (with reversed polarity) and *npn* transistors are the only type encountered in the chapters ahead.

As stipulated in the preceding discussion, proper DC bias conditions must exist in order to achieve the desired performance. Depending upon the DC bias, the transistor will operate in one of the following modes of operation: cutoff, active, or saturation. In the first case, the emitter-base junction is reverse biased which means $V_{BE} < V_{BE(on)}$ for the *pn* junction (0.75v). This also implies that $V_{BC} < V_{BC(on)}$ for the collector-base junction. Therefore, the collector-base junction is also reverse biased. This condition is known as the "cutoff" mode since effectively no current flows through the transistor.

In the two remaining modes, the emitter-base junction is forward biased, and the transistor conducts current. The mode of operation is distinguished by the condition of the collector-base junction — using the emitter as a common reference for both the collector and base. If $V_{CE} < V_{CE(sat)}$ then the base-collector junction is saturated, and the flow of current from collector to emitter is not linearly dependent on I_B. Conversely, when $V_{CE} > V_{CE(sat)}$ for the base-collector junction, then it is reverse biased and current is swept from the collector, across the base, and out of the emitter in linear proportion to the amount of base current applied. This is known as the active region.

Table (2-1) summarizes the relationships which govern the three regions of operation. Furthermore, Figure (2-7) is an *i-v* curve for the Hughes InP HBT (1x1 micron). It serves to illustrate the active and saturation modes of BJT operation while also providing necessary design information that relates the base-emitter voltage drop (V_{BE}) to collector current levels (I_c).

The linearly proportionate increase in collector current relative to base current is referred to as the common-emitter current gain, Beta (β), as shown in Equation (2-5). (Sedra, 1998)

(2-5)
$$\beta = \frac{I_c}{I_B}$$

16

Mode of Operation	Base-Emitter Junction		Collector-Emitter Junction	
	Bias	Relationship	Bias	Relationship
Cutoff	Reverse	$V_{BE} < V_{BE(on)}$	Reverse	--
Saturation	Forward	$V_{BE} > V_{BE(on)}$	Forward	$V_{CE} < V_{CE(sat)}$
Active	Forward	$V_{BE} > V_{BE(on)}$	Forward	$V_{CE} > V_{CE(sat)}$

Table 2-1. Relationships governing the operational regions of the BJT transistors (After Sedra).
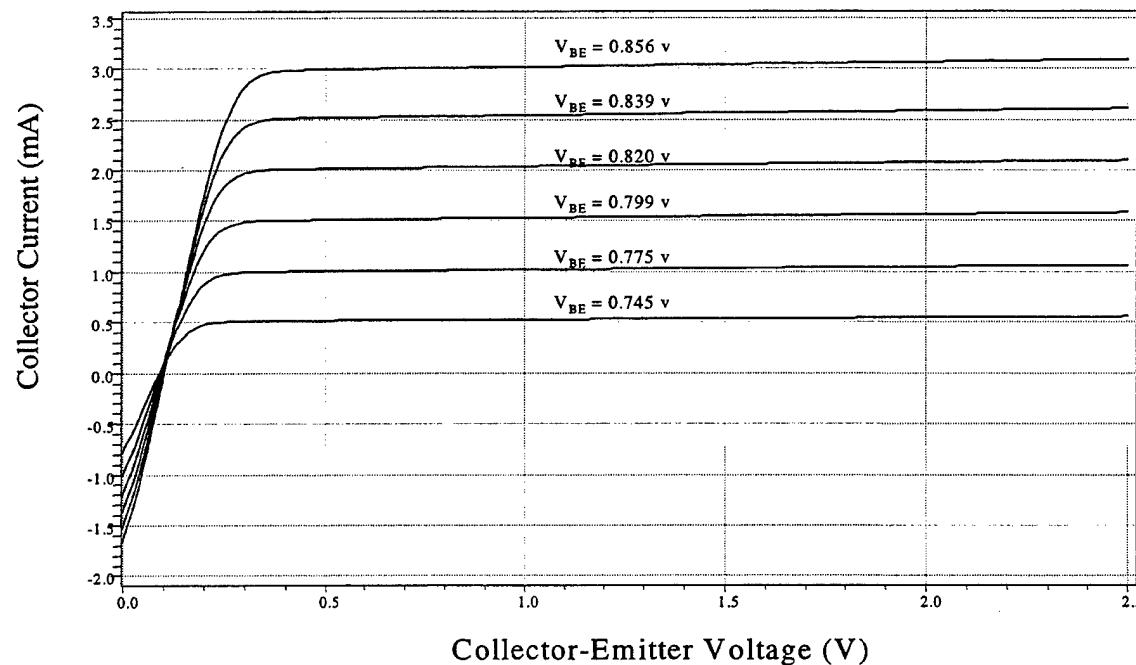
Figure 2-7. I-V Curve for the InP HBT.
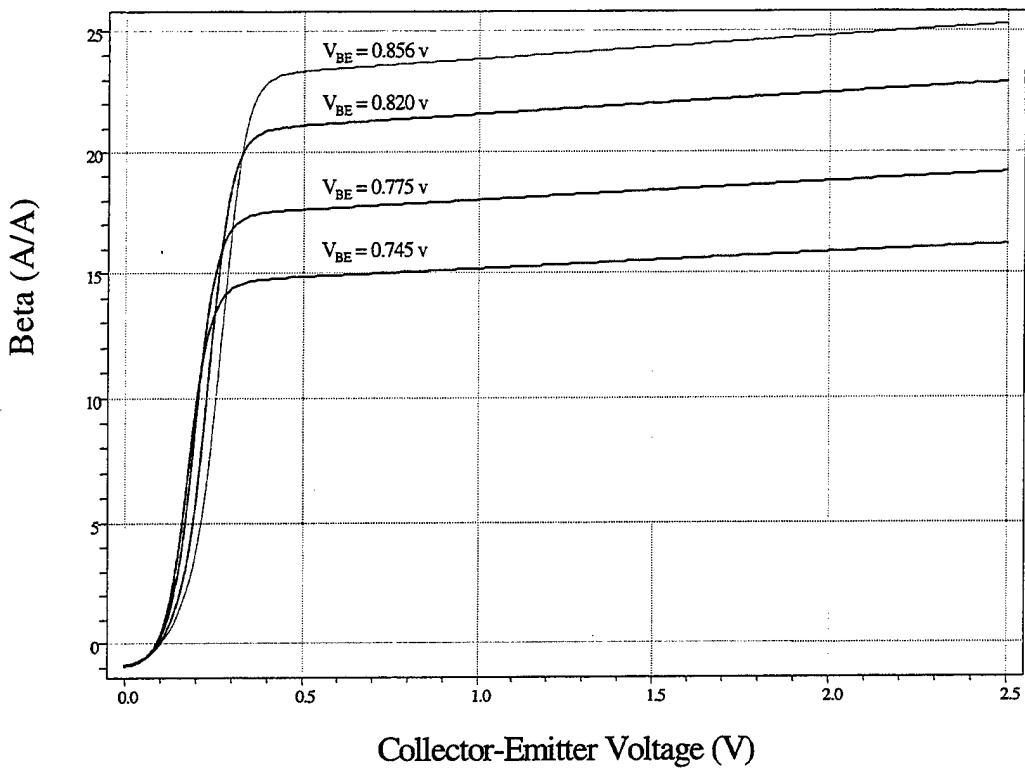
Figure 2-8. **Variation of Beta for the InP HBT with respect to V_{BE} and V_{CE}.**

Beta is a device parameter for BJTs — a function of the device physics and dimensions. Figure (2-8) illustrates how Beta varies according to the values of base-emitter voltage and collector-emitter voltage.

Finally, a simple application of Kirchoff's Current Law produces Equation (2-6) — an important relationship for current through the transistor.

(2-6) $$I_E = I_B + I_C$$

18

c) *DC Analysis of a BJT Circuit*

In order to illustrate the basic concepts of BJT operation as presented in the previous section, the transistor circuit in Figure (2-9) is now examined. Given the reference voltages, the turn-on voltage for the emitter-base junction (0.75v), and Beta for the transistor, it is readily determined that $V_{BE} > V_{BE(on)}$, and therefore the emitter-base junction is forward biased. DC analysis reveals the value of V_B and I_B. Applying the equations from the previous section, I_c, I_E, and V_c are determined, and it is concluded that the transistor is operating in the active region.

DC ANALYSIS:

$V_E = 0v$

$V_B = V_E + V_{BE(on)} = 0.7v$

$I_B = \dfrac{V_{BB} - V_B}{R_B} = \dfrac{5v - 0.7v}{100k\Omega} = 43$

$I_C = \beta \times I_B = 4.3mA$

$I_E = I_C + I_B = 4.343mA$

$V_C = V_{cc} - I_C R_C = 10v - (4.3mA)$
$\qquad = 6.7v$
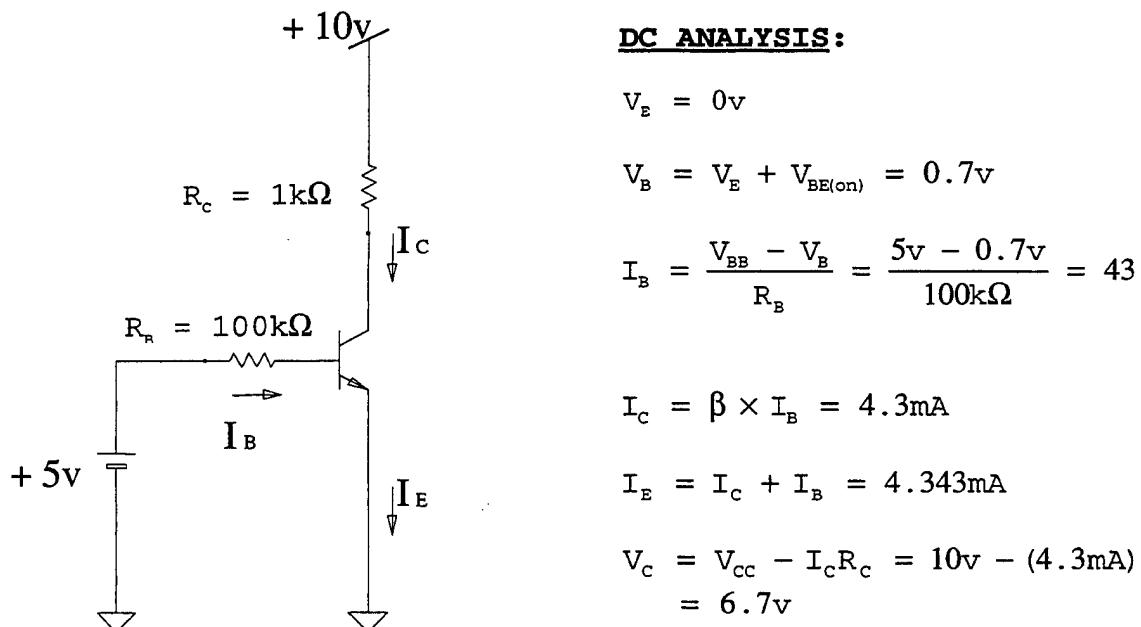
Figure 2-9. DC Analysis of a simple BJT circuit.

In anticipation of logic applications, consider the base voltage as a logical input which is either *high* (above $V_{BE(on)}$) or *low* (below $V_{BE(on)}$). For a logic *high* input the transistor operates in the active mode, causing the voltage at the collector drop below V_{cc} by an amount equal to $I_c R_c$. Alternately, for a logic *low* input the transistor operates in the cutoff mode, drawing effectively no current through the collector and leaving V_c approximately equal to V_{cc}. The functionality of this circuit is essentially that of a basic BJT inverter.

d) BJT Differential Pair

Before committing to the discussion of transistor logic circuits, it is necessary to introduce a configuration that maximizes the switching speed of the BJT transistor: the differential pair. A differential pair is constructed from two matched transistors (Q_1 and Q_2) with their emitters attached to a common current source and their collectors independently biased via separate pull-up resistors to a common voltage source, as shown in Figure (2-10). The base terminals are attached to separate voltage sources of equal value. Assuming the transistors have been given the proper DC bias for operation in the active mode, the relationship in Equation (2-7) is readily determined.

(2-7)
$$I_{E1} = I_{E2} = \frac{I_{bias}}{2}$$
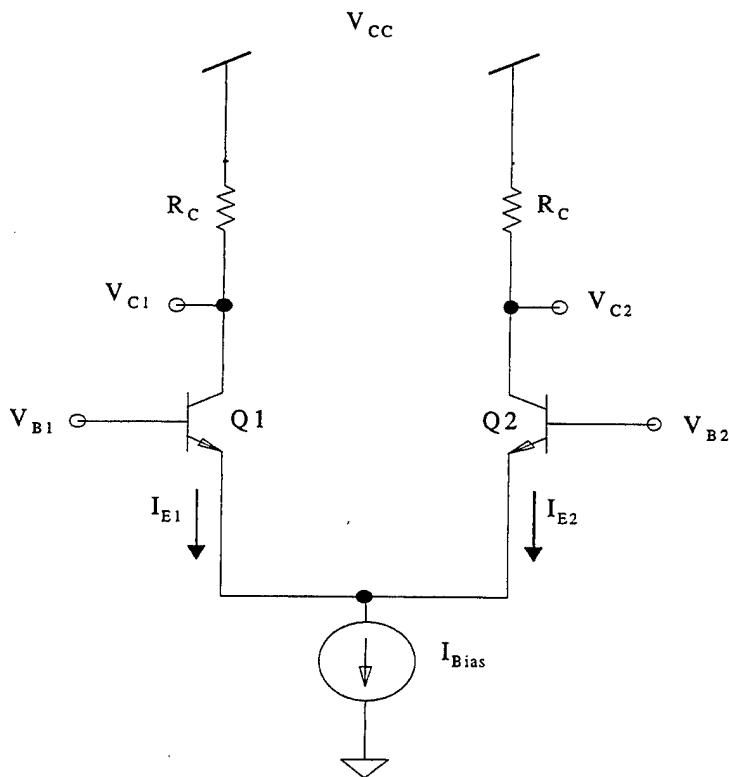
Figure 2-11. Example of a BJT Differential Pair configuration.

Now, consider the scenario where V_{B2} is constant and V_{B1} is allowed to vary between two extremes: one above and one below V_{B2}. When V_{B1} reaches a voltage sufficiently larger than V_{B2}, all of the current from I_{bias} is steered through Q_1 such that Q_2 is cutoff. Conversely, when V_{B1} drops sufficiently below V_{B2}, Q_2 is on and Q_1 is cutoff. As noted in the DC analysis of the previous BJT circuit, the collector voltage of Q_1 exhibits the behavior of a logic inverter with respect to V_{B1}, while the opposite collector voltage (Q_2) functions as a non-inverting buffer.

21

While the availability of complementary output voltages is certainly convenient, the most important observation of the differential pair is its switching speed. A relatively small voltage difference between V_{B1} and V_{B2} is required to switch the current almost entirely to the opposite path. More specifically, for a differential pair implemented with the Hughes InP HBT, it is shown in Figure (2-11) that a difference of only 75mV is sufficient to switch 90% of the current.
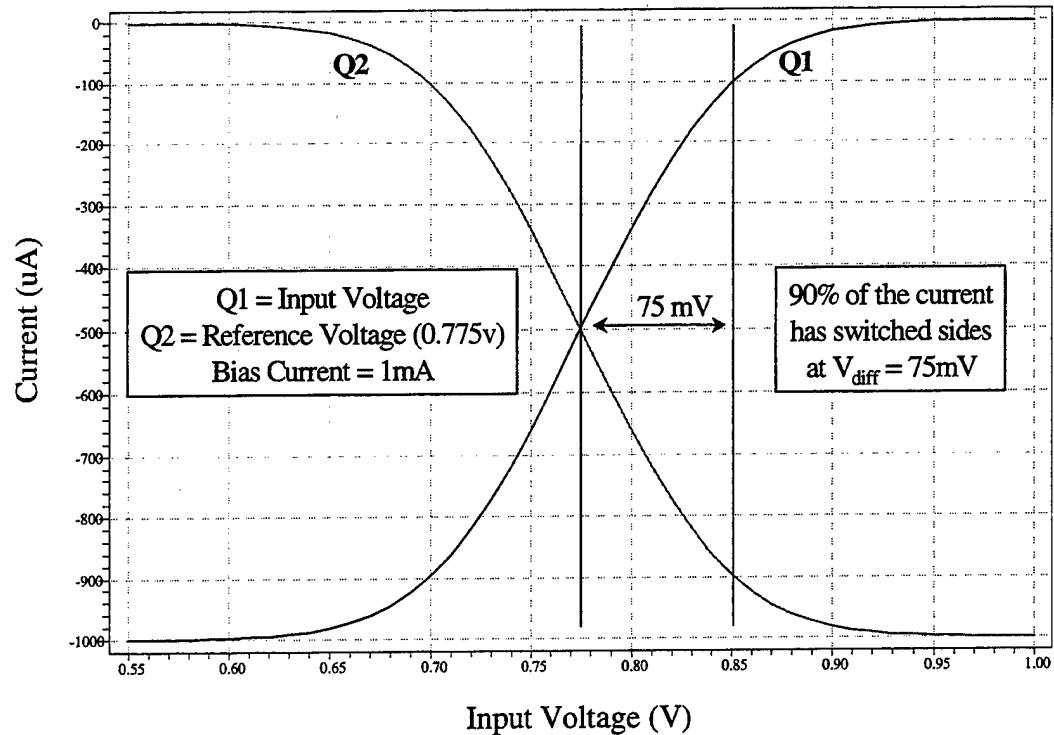
Figure 2-11. Current Switching Characteristic of the InP HBT Differential Pair.

Furthermore, since Q_1 and Q_2 are biased to operate in the active mode, the switching occurs faster than scenarios which may place the transistors in saturation mode. This is because a saturated transistor stores charge in its base. That charge must be dissipated before switching can occur.

It is the current-steering property of the differential pair configuration which ultimately provides a foundation for the development of current mode logic, as will be discussed later in this chapter. However, before reaching that discussion, a brief overview of the dominant BJT logic families will serve to accentuate the advantage of current mode logic.

2. BJT/HBT Logic Families

This discussion is not intended to address all BJT/HBT logic families. Rather, the purpose here is summarize the principles of the two most popular and relevant BJT/HBT logic families. These are transistor-transistor logic and current-mode logic. Ultimately, this discussion culminates with a comparison of the two logic families in order to justify the implementation of current-mode logic for high-speed applications.

a)   *Transistor-Transistor Logic (TTL)*

Transistor-transistor logic evolved directly from diode-transistor logic (DTL) in a successful effort to

23

eliminate the drawbacks of DTL. (Richards, 1967) While there were several stages in this evolution, the end product is a TTL family which resembles the inverter shown in Figure (2-12). The enhanced performance of TTL is predominately achieved through two fundamental design features.

The first improvement is the use of a second transistor in place of the diodes of a DTL circuit. For a
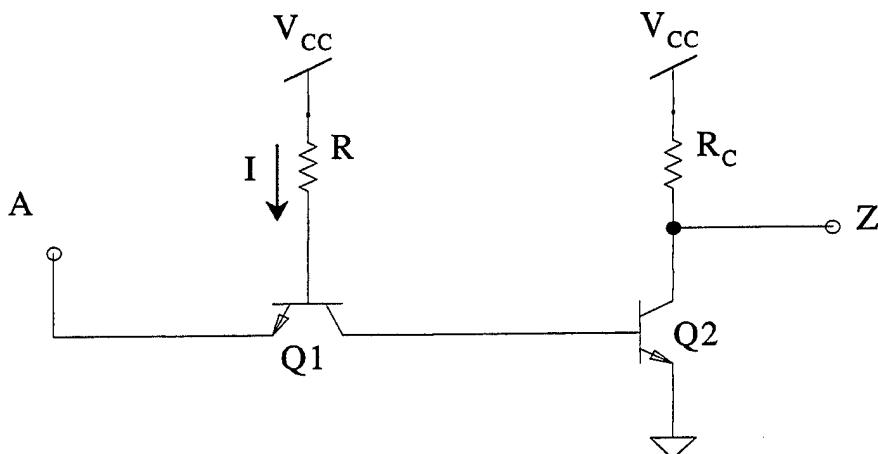
Figure 2-12. TTL Inverter.

low input voltage, Q_1 is turned on — rapidly drawing current from the base of Q_2 and dissipating the excess charge to achieve a faster transition. In the opposite case, when the input is high and Q_1 is cutoff, Q_1 is specifically engineered to have a low reverse Beta such that a small yet sufficient current flows out through the collector and is applied to the base of Q_2.

The second improvement is the use of an optimum output stage, commonly referred to as the "totem-pole"

24

output stage (not shown in the Figure 2-12). It combines the rapid high-to-low transition capability of the common-emitter output stage with the rapid low-to-high transition capability of the emitter-follower output stage.

Based upon these two features in conjunction with other minor modifications, TTL logic achieved a level of popularity which made it the dominant design for SSI, MSI, and LSI circuits throughout two decades. Despite this success, standard TTL circuit speeds are still limited by two design issues. First, transistors operate in saturation mode which increases junction capacitance and its associated switching delay. Second, the resistance along the dissipation path for junction capacitance further increases this delay.

b)   Current-Mode Logic (CML)

Current-mode logic is distinct from the design of other BJT/HBT logic families. The term "current-mode" refers to the channeling of a constant current along alternate paths to achieve logic functionality in circuits. Since it is the presence or absence of current that determines the logical output, the maximum voltage swing can be relatively small in contrast to voltage-mode circuits, such as TTL.

The distinguishing design feature of current-mode logic circuits is the BJT differential pair. It is the

backbone of all CML circuits and the source of critical advantages and disadvantages. The benefit of smaller logic swings has already been mentioned. Also, the discussion of the BJT differential pair earlier in this chapter explained how the collector voltage swings (inverts) rapidly in response to reversing the polarity/magnitude of the differential inputs by a narrow margin of approximately 75mv. This translates into a switching speed for CML which is unsurpassed by its predecessors. Contributing to this remarkable speed is the fact that the transistors of the differential pair can be operated in the active region and, therefore, do not suffer from the effects of excess charge stored at the transistor base. Unfortunately, the constant flow of current which enables these remarkable switching speeds also consumes a remarkable amount of power.

For an illustration of how a CML circuit functions, consider the inverter in Figure (2-13). Let input B have a constant value — a reference voltage. When input A is *high* (greater than the reference voltage by at least 75mv), then Q_1 is turned on and Q_2 is cut off. The current being drawn through R_1 produces a logic *low* ($V_{cc}-I_cR_1$) at V_{out1}. Notably, the complement of this output, a logic *high* (V_{cc}) is simultaneously available at V_{out2}. The presence of complementary outputs is yet another benefit of CML circuits. When input A is switched from *high* to *low*, the
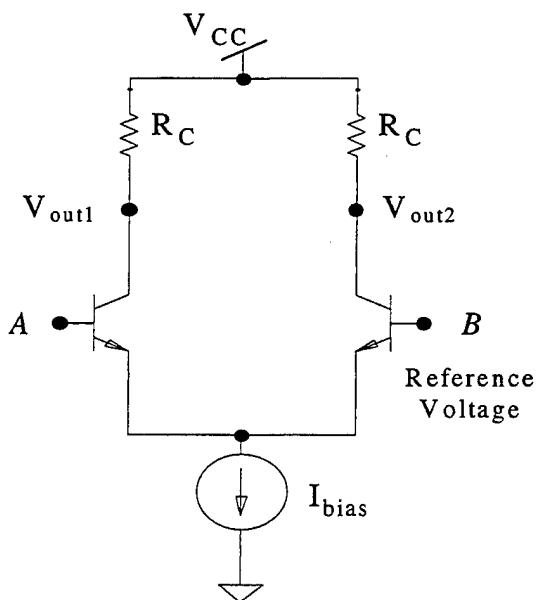
Figure 2-13. CML Inverter.

conditions for Q_1 and Q_2 reverse. Q_2 turns on and Q_1 is cut off. V_{out2} is pulled *low* while V_{out1} is pulled *high*.

c)   *Advantages and Disadvantages*

For high-speed applications, the selection of a BJT logic design is reduced to a quantitative comparison of TTL and CML. The predecessors of these two logic families are far inferior in their capability to dissipate the accumulated charge at the transistor base upon switching.

If the only two criteria were maximizing speed while minimizing power consumption, then there could possibly be a toss-up between TTL and CML — ultimately to

be determined by the design which achieves the lowest power-delay product or by weighting one specification over the other (high-speed or low-power). Clearly, TTL is the low-power contender, while CML is the high-speed champion. However, before addressing the issue in the context of this design project, consider the following summary of advantages and disadvantages.

In addition to being faster, CML requires a smaller voltage swing than TTL and is less susceptible to noise due to the nature of the BJT differential pair. As another benefit of that nature, CML generates complementary outputs. The fact that both output signals are referenced to V_{cc} provides for exceptional stability when V_{cc} is referenced to ground and a negative supply voltage is used. Unfortunately for TTL, its strong point of consuming less power has a down side: the short pulses of current which must be generated for switching logic levels also create spikes in the supply voltage. The constant current drawn by CML circuits avoids this potential source of noise.

In conclusion to this comparison, a logic designer presented with the choice of CML or TTL would only choose TTL in the event that power consumption made CML impractical. In real world applications, this is typically true. However, since it is the purpose of this design project to explore the impact of high-speed logic on digital

28

system architecture, priority has been given to the superior speed and extensive design benefits of CML.

Having concluded that current-mode logic is the best approach to HBT high-speed logic design, it is necessary to design a sufficient set of logic gates to implement the desired test circuit, an 8x8 bit pipelined multiplier. Chapter III presents the discussion of logic circuit design which includes design of the following: an inverter/buffer gate, a NOR/OR gate, full adders, and a practical current source.

THIS PAGE LEFT BLANK INTENTIONALLY

III.   HBT CML LOGIC CIRCUIT DESIGN

A.   DESIGN OVERVIEW

In this chapter, CML logic circuits are designed which will serve as the building blocks for construction of the multiplier logic. The design process is presented in the context of a single logic circuit, beginning with the most fundamental functions and progressing toward the more complex. Of note are the following general design goals which served as guidance for decision-making in the early stages of logic circuit design:

- Minimize the rail voltages (i.e. supply voltage)
- Achieve proper DC bias conditions with reliable noise margins and fanout
- Optimize transient performance for speed and power consumption

B.   INVERTER DESIGN

1.   Circuit Topology

Based upon the introduction to CML design in the previous chapter, Figure (3-1) illustrates the circuit topology of a CML inverter. A detailed description of its function is presented in the previous chapter and will not be repeated here. However, there is one subtle constraint in this design. One of the differential inputs is tied to a
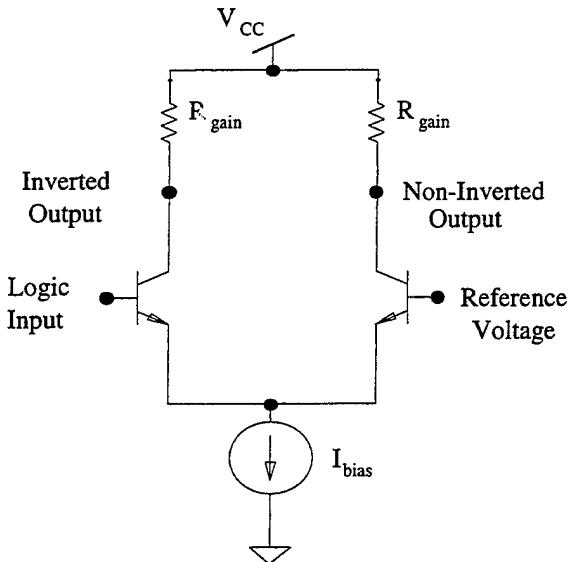
Figure 3-1. CML Inverter.

reference voltage. While this is not essential for the design of an inverter, it will prove significant in the implementation of multiple-input logic gates. A common reference voltage eliminates the need to provide complementary logic signals for each input and furthermore, it avoids the increase in supply voltage associated with multiple complementary inputs in a stacked series of differential input pairs.

Figure (3-2) illustrates the same inverter design as Figure (3-1); however, it also includes an emitter-follower stage at each collector output of the differential pair. The purpose of this stage is twofold. First, it provides a buffer between the input differential pair and the capacitive load of subsequent driven logic gates. Second,
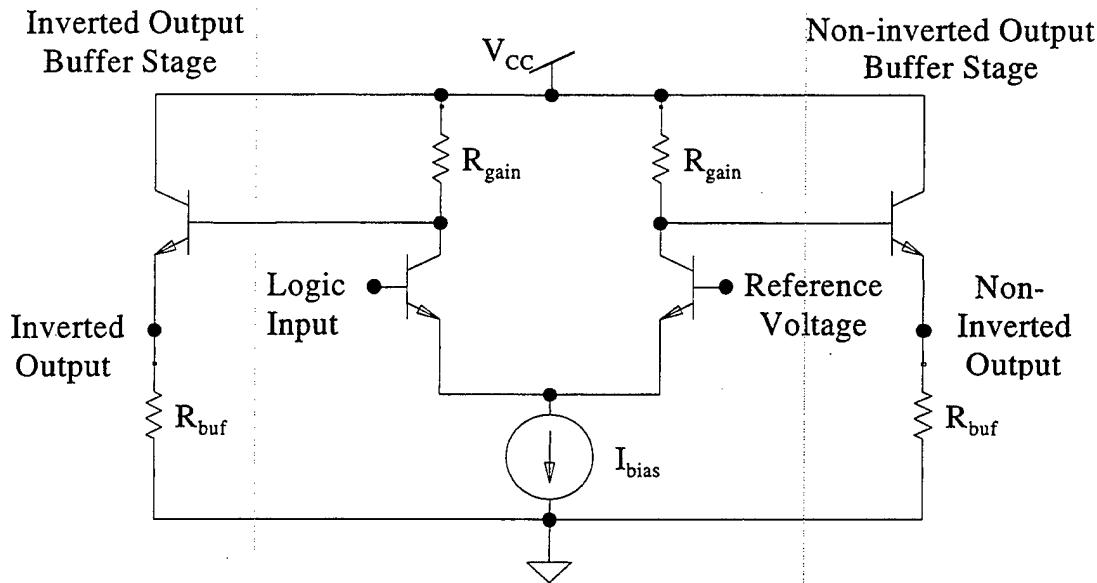
Figure 3-2. CML Inverter with output buffer stages.

it produces a downward DC shift equal to the base-emitter turn-on voltage. Ideally, the gain of the emitter-follower is one; however, in practice the gain is slightly less than one. The result is a slightly diminished voltage swing at the output of the emitter-follower when compared to the voltage swing at the collector of the differential pair.

Whether or not to include the buffer stage represents a fundamental design issue for CML logic circuit design. At a glance, performance arguments can be made both for and against it. On the one hand, it would appear to increase fanout performance, yet on the other, it would appear to decrease switching performance with the additional switching delay of a second transistor stage. Additionally, the non-buffered output topology would consume less power for a

33

given bias current. However, without performance data to substantiate one option over the other, both will be developed and evaluated until objective design considerations can identify a clear preference.

2. Initial Conditions and Design Parameters

a) *Voltage Parameters*

Having introduced the topology of the CML inverter, it is necessary to establish initial conditions for operation. The first is the supply voltage, which is bound by two primary considerations. It must be large enough to support the proper function of the circuit, i.e. provide proper transistor bias conditions and the desired voltage range between high and low logic levels. Conversely, it should be kept as small as possible, because the power consumed by the circuit is directly proportional to the magnitude of the supply voltage.

Clearly, foresight must be exercised in order to determine the minimum supply voltage necessary to achieve proper DC bias conditions for all transistors in all circuits of the design. In the context of this project, the D-type latch design (presented in Chapter IV) imposes the greatest demand on the supply voltage level by operating three transistors in series between the voltage supply rails. For optimum, reliable clocking performance of the latch, the logic reference voltage is determined to be 1.45

volts. This figure is based upon a maximum logic signal range of 0.5 volts and a maximum logic high voltage of 1.7 volts (reference Chapter IV-A-3a for further details).

Given this information, the minimum required supply voltage is determined for each inverter topology. Both require that the voltage at the collector (V_c) be large enough to avoid saturation of Q_1. Furthermore, both require that the voltage at the collector provide for an output voltage that matches the range of the input voltage.

For the non-buffered topology, this implies an inverse match between the voltage at the base of Q_1 and the voltage at its collector. In other words, for a logic input that is high, $V_{B(hi)}$, the output voltage at the collector should be low, such that the following relationship in Equation (3-1) holds true.

(3-1) $$V_{c(low)} = V_{B(hi)} - 0.5v$$

Assuming the collector of Q_1 draws approximately 1mA of current, collector-emitter saturation voltage, $V_{CE(sat)}$, is 0.275 volts and the base-emitter turn-on voltage is 0.775 volts. Under these conditions, Q_1 is on the boundary of active mode operation. For a signal swing larger than 0.5 volts, the transistor would saturate. Conversely, for a logic input (V_B) that is low, the collector voltage (V_c) must be given by Equation (3-2).

(3-2) $$V_{c(hi)} = V_{B(low)} + 0.5v$$

35

For $V_{B(low)}$ equal to 1.2 volts, $V_{c(hi)}$ must be 1.7 volts. Thus, for the non-buffered topology, the maximum voltage at the collector is 1.7 volts. No current flows through R_{gain} because Q_1 is cutoff; therefore, the minimum required supply voltage is also 1.7 volts.

In the case of the buffered topology, the DC voltage drop across the base-emitter junction of the output buffer imposes a greater demand. For the output voltage range to match the input voltage range, the voltage at the collector (as described in Equation 3-2) must be increased by an amount of $V_{BE(on)}$ (as shown in Equation 3-3) in order to counter the base-emitter voltage drop at the buffered output.

(3-3) $$V_{c(hi)} = V_{B(low)} + 0.5v + V_{BE(on)}$$

Assuming a current of 1mA or less through the buffer, $V_{BE(on)}$ is 0.775 volts. The result is a minimum required supply voltage of 2.5 volts. (Reference Chapter IV-A-3a for a thorough derivation of these conclusions.)

In summary, different supply voltage levels will be utilized for the two inverter topologies. The non-buffered output topology will employ a 1.7 volt supply voltage, while the buffered output topology will employ a 2.5 volt supply voltage.

36

b) Transistor Area/Size

In order to optimize switching speeds in BJT/HBT transistors, it is desirable to keep the device area small, thereby minimizing parasitic capacitances. Likewise, a smaller device size requires less current and less current means less power. The InP HBT device sizes made available from Hughes Research Laboratories have junction areas of 1x1, 1x3, 1x5, and 2x5 microns. The 1x1 area transistor is, therefore, the transistor of choice for switching applications (logic circuits). Note, however, that the consideration of device size must be re-visited for applications where switching speed is not a factor, i.e. the construction of a practical current source (addressed in Chapter IV).

c) Fanout Requirement

Fanout is the number of logic gate inputs that a single gate output can drive, while providing voltage levels within the correct logic range. Increased fanout is achieved at the expense of power consumption and loss of speed. Considering that the CML logic inputs/loads are current-driven, increased fanout will require a corresponding increase in switching delay and/or current. As a result, the fanout parameter should be chosen such that it sufficiently economizes the number of logic gates and levels of logic required without needlessly sacrificing

power and speed. In meeting this requirement, a reasonable fanout parameter has been established based upon the logic-level design of the a three-input adder (reference Chapter III-D). For implementation using the minimum number of logic levels, a three-input adder requires a fanout of four.

3. **DC Analysis**

   a) *Overview*

Given the circuit topology for a CML inverter as shown previously in Figure (3-2), the first step in circuit design is to establish the proper DC bias conditions for operation. This can be done for both the buffered and non-buffered cases simultaneously. For the non-buffered case, simply disregard the presence of the buffer stages. The remaining node voltages at the collector outputs on the differential pair are the same.

Figures (3-3a) and (3-3b) show the DC node voltages for the desired operation of a CML inverter given a high logic input and a low logic input, respectively. Given matched transistors the two sides of the differential pair could be considered symmetric in their behavior, except that the input voltages driving the opposite sides of the differential pair are not symmetric. That is, the reference voltage drives the differential pair at 1.45 volts whereas the logic input drives it at 1.7 volts. The result is a difference of 0.25 volts at the emitter. This is a minor
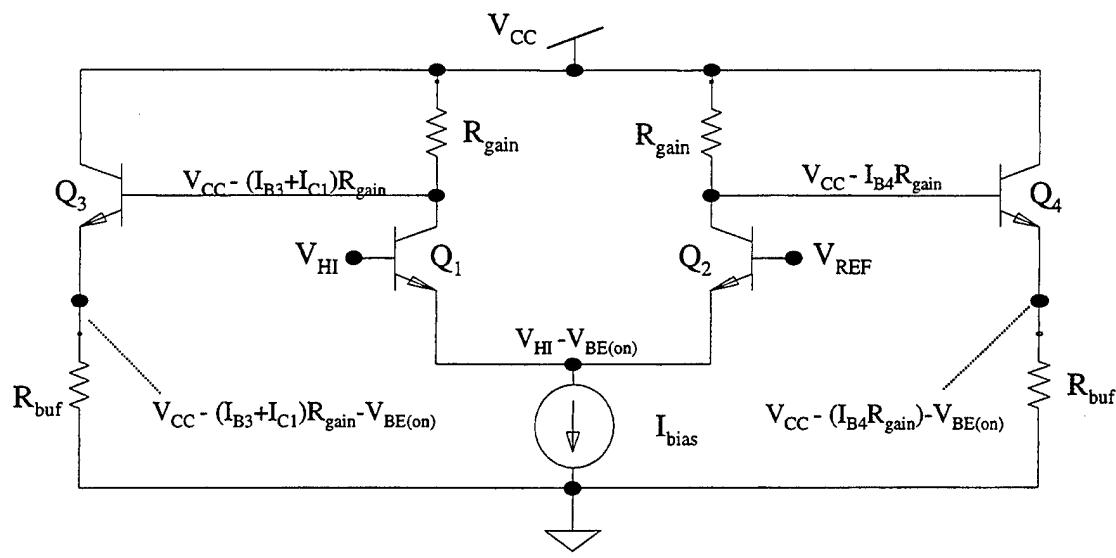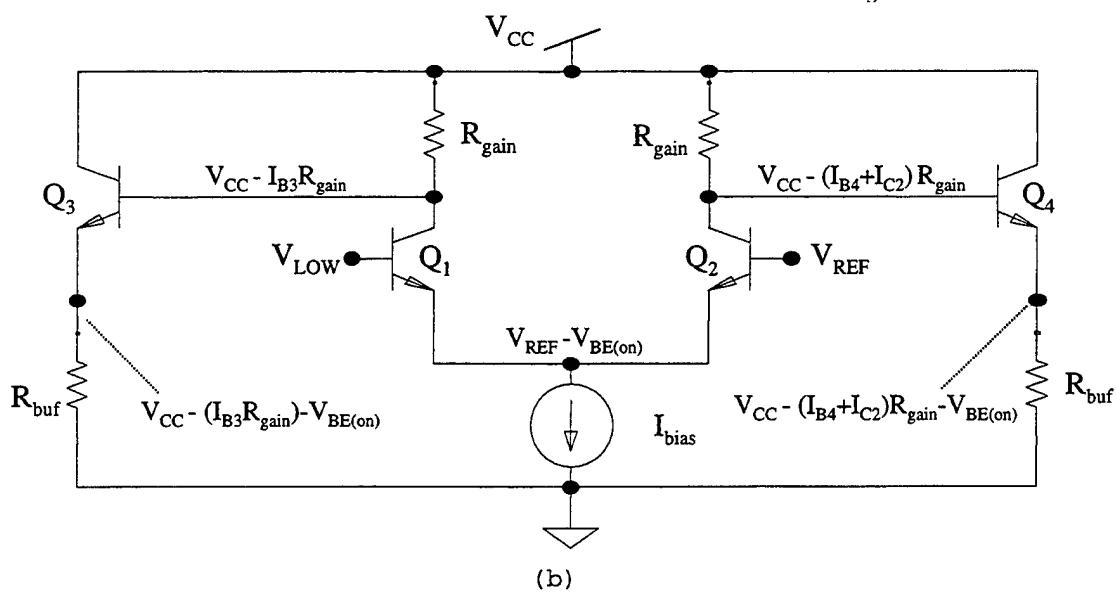
Figure 3-3. DC Analysis of a CML Inverter for (a) a HIGH
input logic level and (b) a LOW input logic level.

observation at present, but it explains the non-symmetric
performance that is encountered between the two output
signals (the inverted and the non-inverted signals).

b) Gain Resistor

In order to take advantage of the switching speed of the differential pair, transistors must be biased to operate in the active mode. Therefore, the value of the base-emitter voltages (V_{BE}) for Q_1 and Q_2 must be such that $V_{CE} > V_{CE(sat)}$. Thus, for a given supply voltage and bias current, there is a restriction on the magnitude of the voltage drop across R_{gain}. If the drop is too large, the transistor will saturate. Conversely, the voltage drop must not be too small because it is the product of I_{R-gain} and R_{gain} which determines the magnitude of the signal voltage swing (assuming active operation). This same voltage range applies to the output of the buffer stages as well. As referenced earlier in this chapter, a constant DC shift of $V_{BE(on)}$ is the only difference between the nodes V_c, and V_{buf}.

In summary, the significance of R_{gain} is two-fold: it must be small enough to keep Q_1 (and Q_2) operating in the active mode, and it must be large enough to provide a satisfactory voltage swing between logic levels. Figure (3-4) illustrates the DC transfer characteristic of the inverter for various values of gain resistance. It effectively demonstrates the upper and lower limitations of gain resistance for a value of I_{bias} equal to 1mA. At resistances of 500 ohms and less, the desired 0.5 volt signal swing is not achieved, and at resistances of 600 ohms
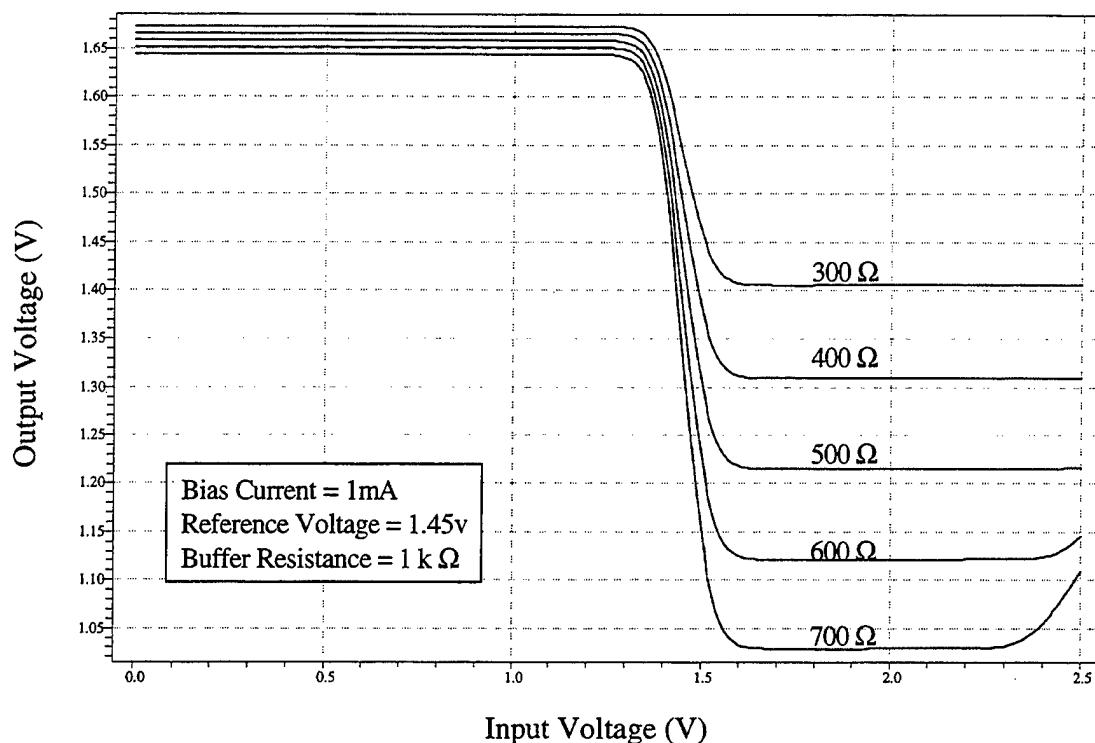
Figure 3-4. Effect of Gain Resistor Variation on Inverter Output.

and greater, the effect of saturation can be observed by the upward bend in the curve.

c) Buffer Resistor

The buffer resistor (R_{buf}) governs the amount of current drawn by the emitter of transistors Q_3 and Q_4. The magnitude of emitter current is directly proportional to the base current which is drawn from the collector of the differential pair. Thus, the base current of the output buffer represents a small portion of the current passing through R_{gain}. In this way, the size of the buffer resistor

41

effectively produces a small DC offset at the buffered output while regulating the amount of current drawn through the buffer stage.

This is significant for two reasons. First, it facilitates optimization of switching speed versus power consumption by providing a mechanism for controlling the amount of current flowing through the buffer stage and therefore, available to drive a logic load. Second, R_{buf} is inversely proportional to a DC voltage offset at the buffered output. The ability to control this offset is especially helpful in matching the output signal swing to the input. Figure (3-5) represents the variation of output voltage for a range of resistor values based upon a bias current of 1mA.

d) Bias Current

Bias current is directly proportional to the current (I_c) drawn through the gain resistor (R_{gain}). Therefore, bias current drives the magnitude of the voltage drop produced in the gain resistor, and this voltage drop corresponds to the maximum signal voltage swing. For this reason, a proper combination of I_{bias} and R_{gain} must be determined to provide the desired 0.5 volt swing. In order to select from an infinite set of current-resistor combinations, a likely set of current-resistor pairs will be identified to represent the practical range of

42
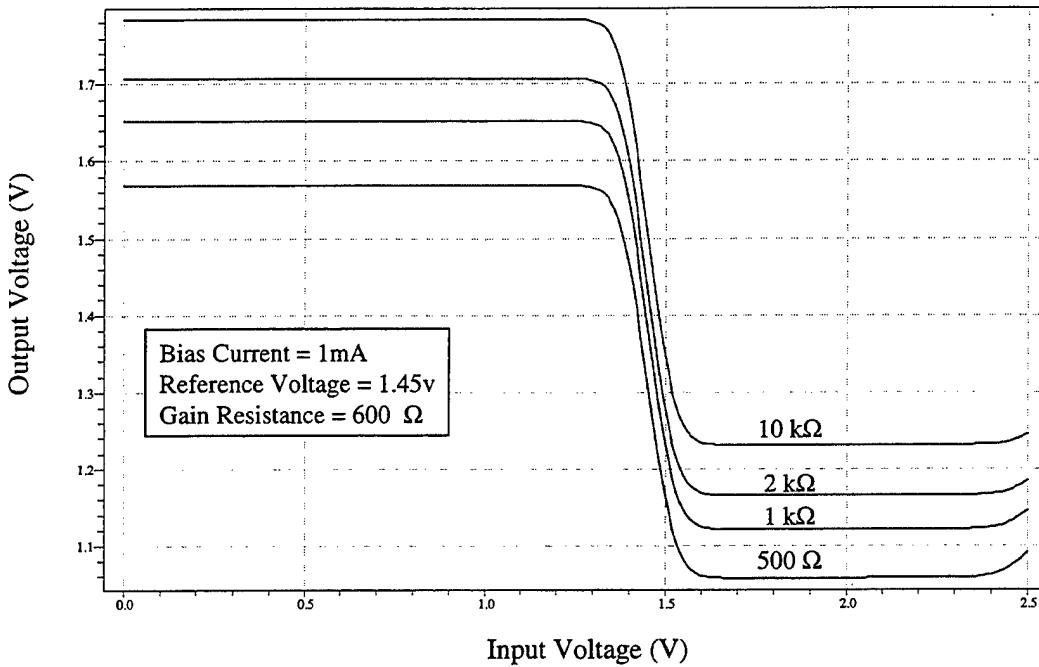
Figure 3-5. Effect of Buffer Resistor Variation on Inverter Output.

possibilities. This is done for both the buffered and non-buffered inverter topologies. Note, the non-buffered topology can be allowed to draw a higher bias current through the differential pair because it does not draw any additional current through buffer stages.

e)   DC Noise Margins

Once values of resistance and bias current are established, the circuit topology is completely defined and a DC transfer curve can be obtained. From this plot the DC noise margins for a particular design are calculated. Noise margins provide a measure of the allowable noise which can

43

be received at the input without affecting the correct logic output. Since this circuit will be operating with such a narrow signal voltage swing, noise margins are a critical interest for establishing reliable DC bias conditions. Equations (3-4) and (3-5) define the high and low noise margins in terms of the maximum and minimum, high and low logic values. (Weste, 1993)

(3-4) $$NM_L = |V_{ILmax} - V_{OLmax}|$$

(3-5) $$NM_H = |V_{OHmin} - V_{IHmin}|$$

where, V_{IHmin} = minimum HIGH input voltage
V_{ILmax} = maximum LOW input voltage
V_{OHmin} = minimum HIGH output voltage
V_{OLmax} = maximum LOW output voltage

These logic values are extracted from the DC transfer curve. The two unity gain points (where the slope equals negative one) of the DC transfer curve have been used to define the boundaries of these regions.

f)   DC Bias Optimization

Given a set of practical current values, DC analysis is employed to identify a set of matching gain resistances which properly bias the inverter for logic operations. For each pair of current-resistor values, a DC transfer characteristic is obtained to determine the noise margins and the maximum range of the signal swing. The results are tabulated in Table (3-1). In the absence of a

Inverter Topology	Bias Current (mA)	Supply Voltage (Volts)	Gain Resistor (Ohms)	Buffer Resistor (Ohms)	Unloaded Low Noise Margin (Volts)	Unloaded High Noise Margin (Volts)	LOADED Low Noise Margin (Volts)	LOADED High Noise Margin (Volts)	LOADED Logic Signal Range (Volts)
NON-Buffered Output	0.1	1.7	5000	n/a	0.11	0.16	0.12	-0.04	0.27
	0.25	1.7	2000	n/a	0.11	0.15	0.11	-0.01	0.31
	0.5	1.7	1000	n/a	0.11	0.14	0.12	0.01	0.34
	0.75	1.7	700	n/a	0.14	0.13	0.14	0.01	0.34
	1	1.7	500	n/a	0.11	0.13	0.11	0.02	0.37
	1.5	1.7	340	n/a	0.1	0.12	0.1	0.02	0.38
	2	1.7	250	n/a	0.09	0.11	0.09	0.01	0.39
	3	1.7	170	n/a	0.08	0.09	0.08	0	0.41
Buffered Output	0.1	2.5	5000	5000	0.11	0.11	0.11	0.1	0.4
	0.25	2.5	2400	5000	0.16	0.16	0.15	0.15	0.51
	0.5	2.5	1200	5000	0.13	0.18	0.13	0.17	0.52
	0.75	2.5	750	2000	0.17	0.13	0.13	0.13	0.52
	1	2.5	600	2000	0.17	0.14	0.17	0.12	0.53
	1.5	2.5	400	2000	0.14	0.13	0.14	0.11	0.53

Table 3-1. Results of DC Analysis on Inverter Configurations

load, each configuration met the established design requirements — that is, a matched input and output signal voltage range of 0.5 volts, centered at a reference voltage of 1.45 volts with sufficiently balanced noise margins of 0.1 volt minimum (20% of the signal range).

However, when examined under the maximum fanout load (which is four), the performance of the non-buffered output topology suffers greatly. The maximum high logic voltage is reduced by an amount ranging from 0.09 volt to 0.23 volt, depending upon the bias configuration. Not only does a load reduce the desired 0.5 volt signal range, but it also erodes the high-end noise margin. As a result, the non-buffered output topology can now be eliminated from further consideration in the design process.

As for the buffered output topology, the noise margins and voltage range are remarkably consistent — regardless of the loading. The output buffer effectively isolates the current drawn by the load from the current in the differential pair. Thus, each of the bias configurations for the buffered output topology will be further tested under transient conditions to identify the optimum inverter design. It should be noted that the DC analysis presented here and the transient performance analysis which follows are both conducted using ideal current source models.

4. AC/Transient Analysis

a) *Delay Measurements*

Transient performance of logic circuits is generally quantified by measuring the delay associated with signal propagation. The delay times utilized here are standard performance parameters. However, for completeness, their mathematical definitions are provided below in Equations (3-6) and (3-7). (Weste, 1993)

(3-6) t_{fall} = time for a logic signal to traverse from 0.9 V_{RANGE} to 0.1 V_{RANGE}

(3-7) t_{rise} = time for a logic signal to traverse from 0.1 V_{RANGE} to 0.9 V_{RANGE}

where, V_{RANGE} = the voltage difference between the steady state V_{HI} and V_{LOW}

b) *Performance Parameters*

At this point in the design process, two performance parameters are of primary concern, power and speed. Being related to each other, there is often a trade-off between the two. Optimization of these two parameters will determine which of the DC bias inverter configurations will be implemented. A common method of optimization is to quantify the parameters of power and speed as a single

figure of merit, such as a product or a ratio. Optimization is then achieved by maximizing or minimizing the appropriate figure of merit.

Power-delay product is one such figure of merit. It is simply the product of the power consumed by a logic circuit multiplied times the propagation delay of the signal from input to output. Expectedly, the design that most efficiently balances the trade-off between speed and power consumption will yield the lowest power-delay product in transient testing.

The ratio of speed to power provides a similar figure of merit, but speed measurements are not as clearly defined as delay measurements. Therefore, in the interest of optimizing this design for speed, a definition of maximum switching frequency will now be established. The maximum reliable frequency is defined as the maximum switching frequency of the logic input signal for which a maximally loaded output signal consistently traverses 90% of the 0.5 volt range of logic.

c)   Transient Analysis Procedures

For an accurate evaluation of logic circuit performance, it is necessary to provide a realistic input signal and a worst-case output load. Here, the term load implies driving four inverters in parallel. To achieve a realistic test environment, the test circuit of Figure (3-6)

was designed. Specifically, note the location of gates *A* and *B*. Their input and output signals will be measured to analyze performance with a fanout of one and four, respectively.
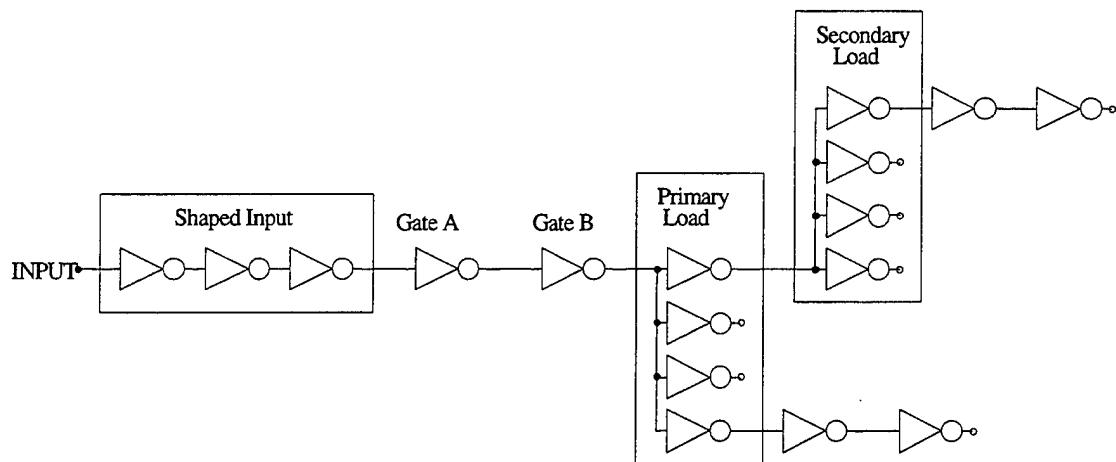
Figure 3-6. Test Circuit for Transient Analysis.

It is expected that the use of a reference voltage at the differential input of the inverter will cause the inverted and non-inverted output signals to respond differently. As a result, two gate topologies are analyzed for each of the valid DC bias configurations from Table (3-1). The first gate topology is a single output inverter from which the inverted output signal is measured. The second is a complementary output inverter from which the non-inverted output signal is measured. Conveniently, these two configurations also represent the alternating signal

49

pattern which will characterize the adder circuits later in this chapter.

Initially, the appropriate logic delays are measured at gate A and gate B in order to collect data for the cases of minimum and maximum loads, respectively. The worst-case delay is then multiplied by the average power per gate to obtain a power-delay product. This is done for both the inverted and the non-inverted output signals — providing separate power-delay product terms. Their sum forms a composite power-delay product. The composite power-delay product is a figure of merit which effectively represents the implementation of the two gate topologies in series.

Finally, the switching period of the input logic is decremented for successive tests in order to determine the shortest period for which the output signal of a loaded gate (gate B) would consistently traverse the full range of logic (between *high* and *low*). This quantity has been defined in the previous section as the maximum reliable frequency (MRF). For each configuration, the maximum reliable frequency is divided by the average power per gate to obtain a speed-power ratio (GHz/mW). The presence of a secondary load provides confirmation that consecutive loads can be successfully driven when the primary load is driven at its maximum reliable frequency.

d) Summary of Results

Transient analysis confirms the non-symmetric behavior of the inverted and non-inverted output signals. Therefore, Tables (3-2a) and (3-2b) provide details of their

Bias Current (mA)	Tprop L-H (pS)	Tprop H-L (pS)	Current per Gate (mA)	Power per Gate (mW)	Maximum Power-Delay Product (mW-pS)
0.1	42	255	0.81	2.03	518
0.25	56	48	0.97	2.42	136
0.5	33	26	1.28	3.20	106
0.75	23	26	1.59	3.99	104
1	17	26	1.88	4.69	122
1.5	13	27	2.38	5.94	160

Table 3-2a.   Power-Delay Data for the Inverted Signal.
Single output topology with practical current sources and a fanout load of four.

Bias Current (mA)	Tprop L-H (pS)	Tprop H-L (pS)	Current per Gate (mA)	Power per Gate (mW)	Maximum Power-Delay Product (mW-pS)
0.1	212	82	1.45	3.63	770
0.25	61	88	1.64	4.10	361
0.5	27	63	2.02	5.04	318
0.75	23	46	2.31	5.78	266
1	19	41	2.63	6.56	269
1.5	18	40	3.09	7.74	309

Table 3-2b.   Power-Delay Data for the Non-Inverted Signal.
Complementary output topology with practical current sources and a fanout load of four.

51

respective delay measurements. Specifically, the high-to-low transition of the non-inverted output signal represents the worst-case transition.

The overall performance of each DC bias configuration is summarized in Table (3-3). The power-delay product and speed-power ratio are normalized to simplify comparison. Figure (3-7) illustrates the minimization curve for the power-delay product, while Figure (3-8) shows the maximization curve for the speed-power ratio.

Clearly, the 0.75mA configuration proves to be the optimum design — maximizing the speed-power ratio while minimizing the power-delay product. Furthermore, it provides for a maximum reliable frequency of 8.7 GHz. This is more than suitable to achieve the 5 GHz maximum clock frequency desired in Chapter V (for the maximally pipelined multiplier implementation).

Bias Current (mA)	Maximum Composite Power-Delay Product	Normalized Composite Power-Delay Product	Maximum Reliable Frequency (GHz)	Normalized Speed-Power Ratio
0.1	467	3.48	n/o	n/a
0.25	144	1.34	5.30	0.86
0.5	96	1.14	7.10	0.94
0.75	**72**	**1.00**	**8.70**	**1.00**
1	67	1.06	9.09	0.92
1.5	67	1.27	11.10	0.96

Table 3-3. **Summary of Transient Analysis Results.** Composite Power-Delay Product and Speed-Power Ratio.
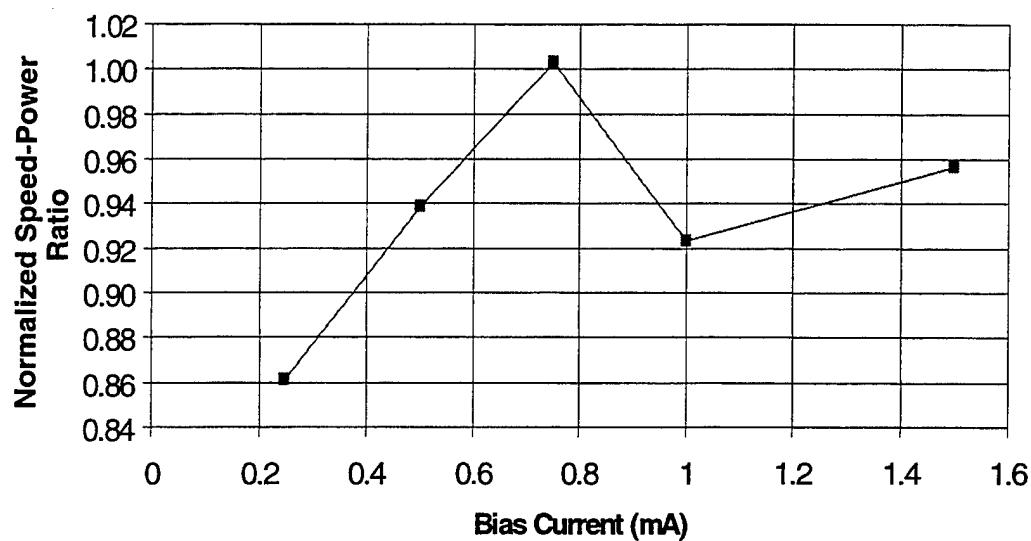
Figure 3-7. Results of Transient Analysis: Normalized Speed-Power Ratio of Inverter Configurations.
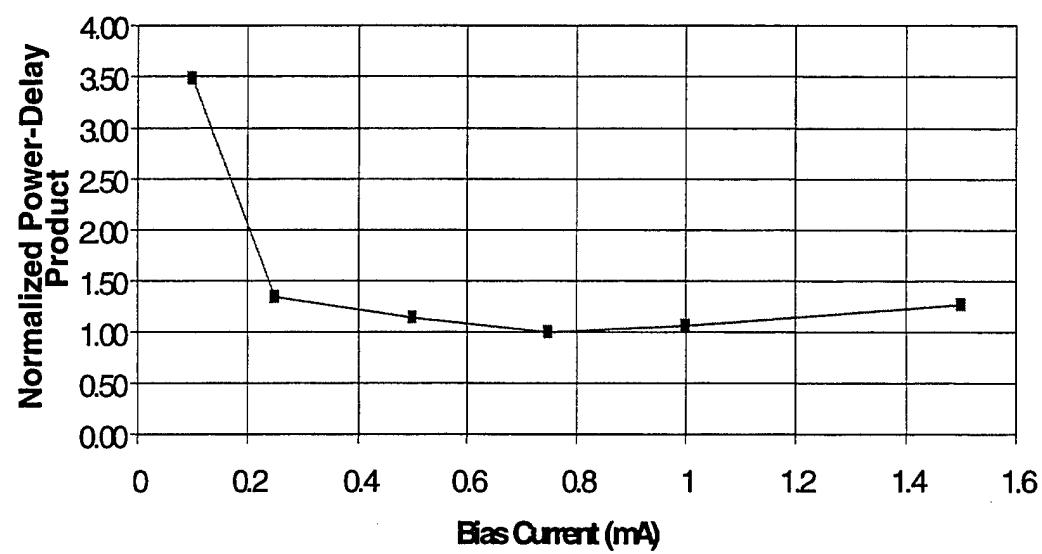
Figure 3-8. Results of Transient Analysis: Normalized Power-Delay Product of Inverter Configurations.

5. Final Design Summary: Inverter

The final design for the CML inverter/buffer circuit is illustrated in Figure (3-9). The applicable design and performance parameters have been summarized in Table (3-3). Here, the data represents performance when the design is implemented with the 0.75mA practical current source from Chapter III-E. Also note that when complementary output signals are not required, the unused output buffer stage can be excluded to conserve power and minimize the device count.

CML Inverter
Design and Performance Parameters

R_{gain}:	750 Ω
R_{buf}:	2000 Ω
I_{bias}:	0.75 mA
NM_L:	0.13v (26% V_{swing})
NM_H:	0.14v (28% V_{swing})
Power:	5.78 mW (complementary output)
	3.99 mW (single output)

Delays	Inverted Signal		Non-inverted Signal	
	Fanout = 1	Fanout = 4	Fanout = 1	Fanout = 4
$t_{p(H-L)}$	14ps	26ps	39ps	46ps
$t_{p(L-H)}$	17ps	23ps	18ps	23ps
t_{fall}	19ps	41ps	87ps	90ps
t_{rise}	48ps	61ps	45ps	60ps

Table 3-4. CML Inverter Design and Performance Parameters.

54

2.5 volts

750 Ω   750 Ω

HBT 1x1

HBT 1x1

V_{LOGIC} HBT 1x1   HBT 1x1   1.45 v

HBT 1x1

Inverted
Output

Buffered
Output

2 kΩ   0.75mA   2 kΩ
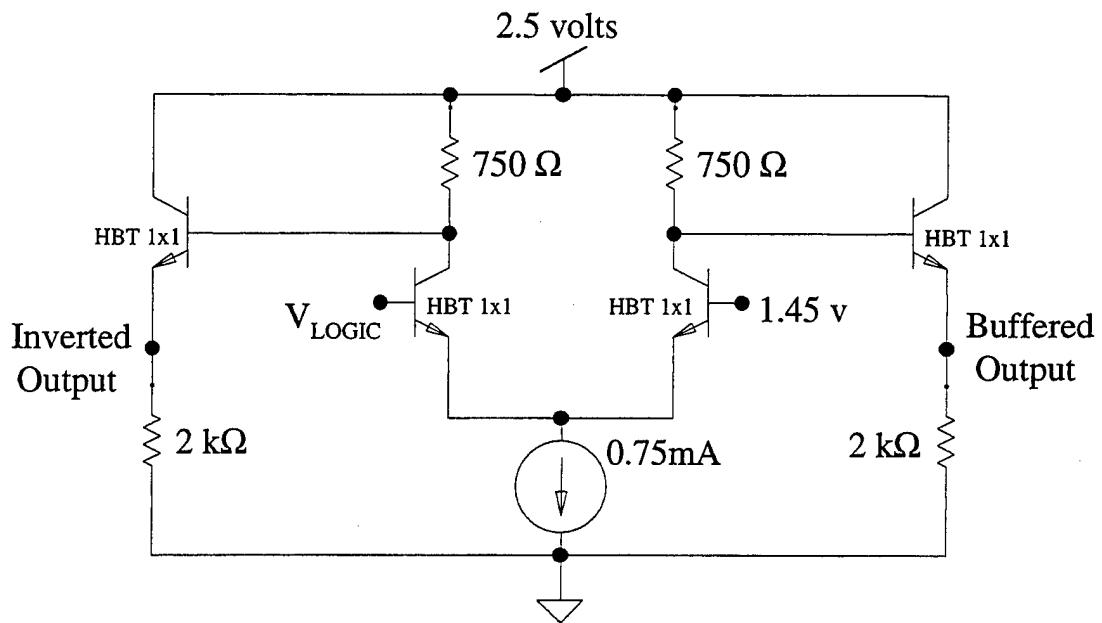
Figure 3-9. Final Design of the CML Inverter.

C.   LOGIC *NOR* GATE DESIGN

1.   Overview and Analysis

The circuit topology for a two-input CML NOR gate is presented in Figure (3-10). There is little that differs from the inverter, which accurately suggests that the analysis here will be extremely similar to the previous section. In fact, with regard to both circuit topology and performance analysis, the only distinguishing feature is the second logic input in parallel with the first.

Consider the functionality of the two parallel inputs A and B. If either of them is a logic *high*, then the left side of the differential pair is on and the NOR output is pulled *low*. Conversely, if both inputs A and B are *low*,
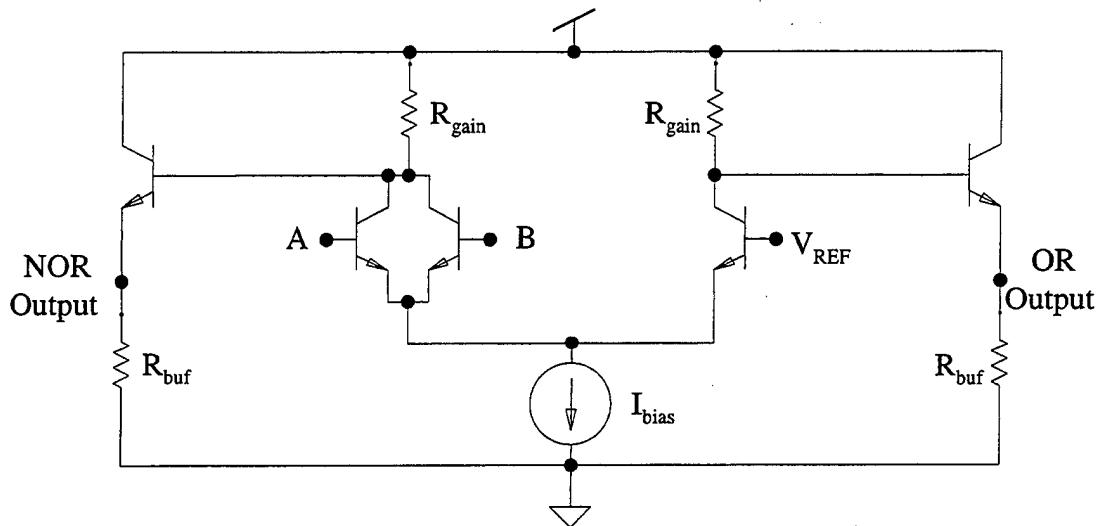
Figure 3-10. Circuit topology for a two-input OR/NOR logic gate.

then the NOR output is *high*. On the opposite side of the differential pair is the complementary output — the OR function. If another input transistor were added in parallel to the existing two, it would be a three-input OR/NOR gate — and similarly for a fourth input.

Despite the drastic change in functionality, the presence of several logic inputs in parallel to the original logic input induces no fundamental change to the DC bias of

the circuit. As a result, the DC bias conditions for the optimized inverter circuit are directly applied to the final design of the NOR circuit.

2. Final Design Summary: *OR/NOR*

With the exception of having multiple parallel transistors for multiple logic inputs, the final design for the CML OR/NOR logic circuit is identical to that of the inverter. As for its performance, the noise margins and delay measurements vary only slightly in response to the "multiple trigger" effect of simultaneous parallel inputs. The design parameters are identical to the inverter and therefore are not repeated. However, a selection of the performance parameters have been provided in Table (3-5) in order to demonstrate the variation of performance based upon the input configuration.

Conveniently, the NOR gate constitutes a near identical capacitive load as the inverter — with maximum delay differences of less than 1.5ps. It exhibits the same delay variations between its OR and NOR signals as the inverter does between the inverted and non-inverted signals. And finally, as with the inverter, when both of the complementary outputs of the OR/NOR gate are not required, the unused output buffer stage is not included to conserve power and minimize the device count.

57

CML OR/NOR Gate
Delay Performance Parameters

2-Input OR/NOR Gate
Single Input Transition

		NOR Signal		OR Signal	
		Fanout = 1	Fanout = 4	Fanout = 1	Fanout = 4
Single	$t_{p(H-L)}$	16ps	29ps	40ps	47ps
Input Transition	$t_{p(L-H)}$	24ps	29ps	19ps	23ps

3-Input OR/NOR Gate
Single and Simultaneous Input Transitions

		NOR Signal		OR Signal	
		Fanout = 1	Fanout = 4	Fanout = 1	Fanout = 4
Single Input Transition	$t_{p(H-L)}$	19ps	28ps	41ps	48ps
	$t_{p(L-H)}$	29ps	34ps	18ps	23ps
Simultaneous Input Transition	$t_{p(H-L)}$	17ps	36ps	40ps	47ps
	$t_{p(L-H)}$	43ps	48ps	11ps	16ps

4-Input OR/NOR Gate
Single Input Transition

		NOR Signal		OR Signal	
		Fanout = 1	Fanout = 4	Fanout = 1	Fanout = 4
Single Input Transition	$t_{p(H-L)}$	21ps	30ps	41ps	48ps
	$t_{p(L-H)}$	33ps	39ps	18ps	23ps

Table 3-5.   Summary of OR/NOR Gate Delay Performance.

3. Implementation of the *AND* Function

In current-mode logic, the AND function is implemented by simply inverting the input signals and reversing the polarity designation of the output nodes. In actual practice, inverters and OR/NOR gates are sufficient to realize any logic function. Thus, for the sake of simplicity, AND gates were not constructed as a separate logic circuit. Rather, all logic functions were deliberately expressed as functions of inverters and OR/NOR gates.

D. ADDER DESIGN

1. Implementation

Two-input and three-input adders are required to construct the carry-save adders and carry-completion adders of the multiplier (Chapter V). Equipped with a sufficient set of logic gates, this is an elementary task. The sum of min-terms for the sum and carry bits of a two-input adder are shown in Equations (3-8) and (3-9), respectively.

$$\text{(3-8)} \qquad \text{Sum}\big|_{2\text{input}} = XY' + X'Y$$

$$\text{(3-9)} \qquad \text{Carry}\big|_{2\text{input}} = XY$$

Employing De'Morgan's Theorem, these expressions can be manipulated into the equivalent expressions for

implementation with OR/NOR gates, as shown in Equations (3-10) and (3-11).

(3-10) $\qquad \left. \text{Sum} \right|_{\text{2input}} = (X'+Y)' + (X+Y')'$

(3-11) $\qquad \left. \text{Carry} \right|_{\text{2input}} = (X'+Y')'$

This adder design requires the complementary logic inputs be provided in order to eliminate the need for inverters and a third level of logic delay. Such a requirement is trivial because complementary signals are potentially available at the output of each CML logic gate. Figure (3-11) illustrates the two-input adder.
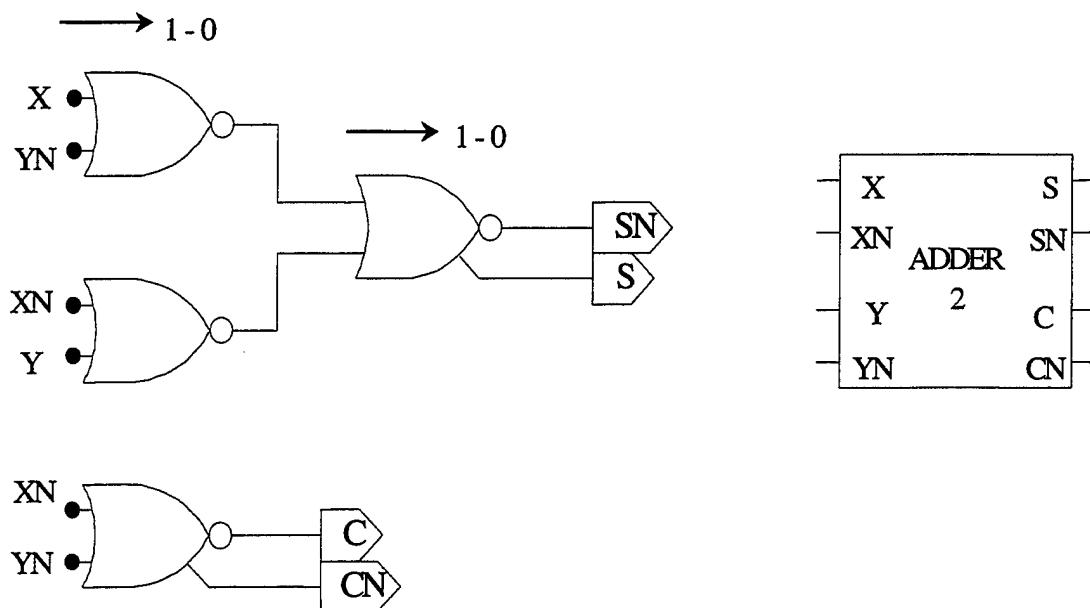
Figure 3-11. Two-input adder with identification of the critical path.

60

A similar procedure was followed to implement Equations
(3-12) and (3-13) for the construction of a 3-input adder,
as illustrated in Figure (3-12).

(3-12) $Sum|_{3input} = (X'+Y+Z)' + (X+Y+Z')'$

$+ (X+Y'+Z)' + (X'+Y'+Z')'$

(3-13) $Carry|_{3input} = (Y'+Z')' + (X'+Z')' + (X'+Y')'$
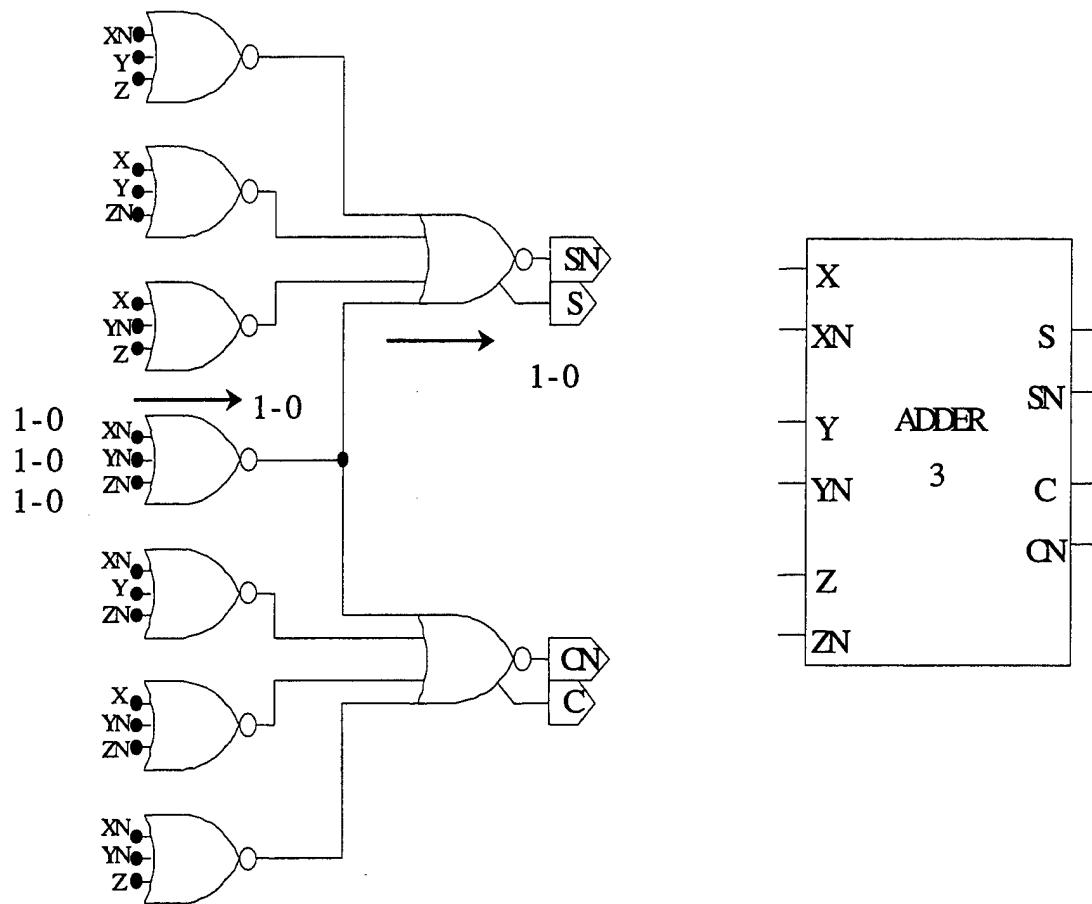
**Figure 3-12.   Three-input adder with identification of
the critical path.**

61

2. Performance Analysis

Proper functioning of each adder was verified for all possible input combinations. Notice that the critical path for each adder is identified in Figures (3-11) and (3-12). For the two-input adder, the critical path flows through two levels of logic to produce the sum bit. The worst case transition is from a (1/0) or a (0/1) input for (X/Y) to a (1/1) input. This is owing to the fact that the worst-case gate delay is the high-to-low transition of the OR output when it has been driven by the high-to-low output transition of the preceding NOR gate. Based upon the data from Table (3-5), the critical path delay equals 63 picoseconds. This provides a good match with a simulation of the critical path delay which yields 60 picoseconds.

Similarly, for the three-input adder the critical path delay is calculated to be 67 picoseconds along the path illustrated in Figure (3-12). This was validated with a simulation measurement of 66 picoseconds.

E. PRACTICAL CURRENT SOURCE DESIGN

1. Circuit Topologies

Up to this point, each logic element has been designed using an ideal current source. In order to validate the performance of these designs for actual implementation, it is necessary to construct a practical current source. There are effectively three circuit configurations which provide

transistor bias conditions for establishing a current source. These three topologies are presented in Figure (3-13). In each configuration the amount of bias current drawn is regulated by and directly proportional to the magnitude of the current drawn by the base of Q_{SOURCE}.
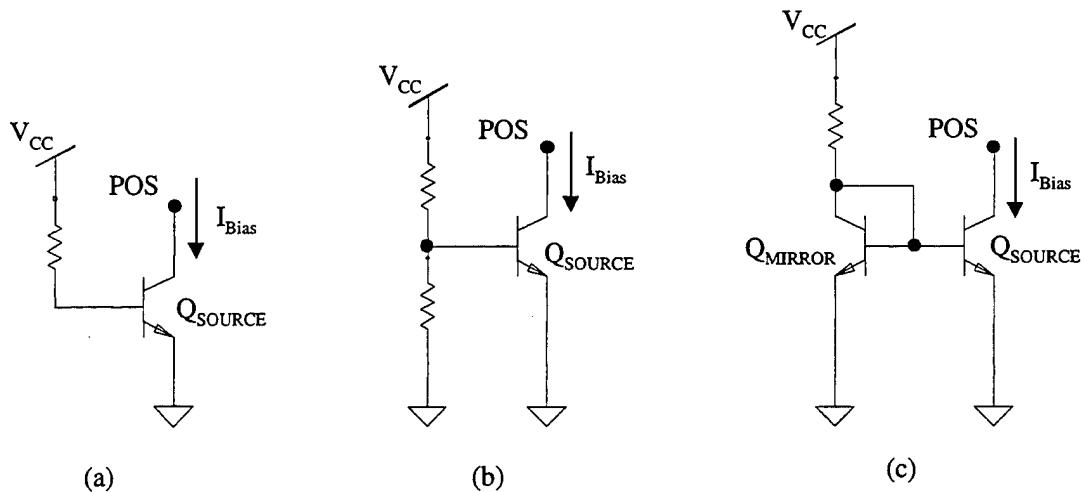
Figure 3-13. Current Source Topologies.

2. Performance Analysis

In order to analyze and compare the performance of each current source, three simple 0.75mA current sources are designed — one using each topology. Each is then implemented as the practical current source for the inverter/buffer circuit of Chapter III-B-5. Their relative performance is evaluated based upon the following design goals:

- Minimize the operational limitations due to frequency response

- Approximate the performance of an ideal current source

- Minimize the cost of implementation (power and device count)

The performance of each configuration is illustrated in Figure (3-14a) and (3-14b). Notice that each inverted output signal drops below the desired 1.2 volt voltage *low* level when making the transition from high-to-low. This "dip" results from reversing the polarity of the differential pair input signals — inducing a brief drop in the bias voltage at the positive (POS) terminal of the current source. A delayed return to the proper bias voltage is then governed by the RC characteristics of the Q_{SOURCE} collector. This delay is particularly observed in the transient performance of the topologies in Figure (3-13a) and (3-13b).

3. Final Design: Current Source

By process of elimination, the current mirror topology of Figure (3-13c) is the only design suitable for driving a logic device family that is capable of switching frequencies above 8 GHz. Unfortunately, the current mirror also incurs the largest cost in terms of power and device count. Thus, to reduce the amount of current "lost" through the left side

64

of the current mirror, Q_{MIRROR} is given a smaller area than Q_{SOURCE}. Testing a variety of such configurations yields a current mirror configuration that implements Q_{MIRROR} with a (1x1) micron transistor and Q_{SOURCE} with a (1x3) micron transistor.
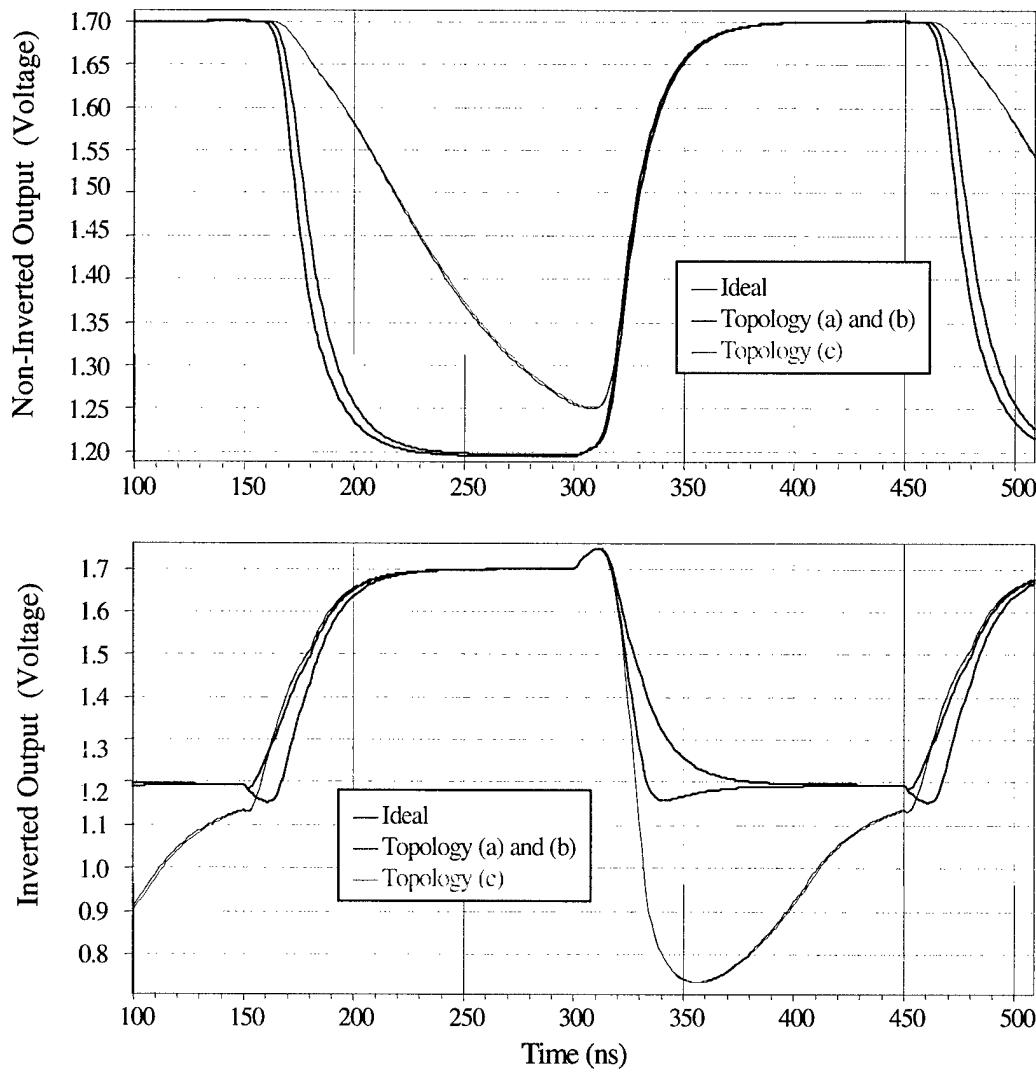
Figure 3-14. Transient performance of three practical current source topologies compared to an ideal source.

65

a) 0.75mA Current Source

The final current source design for a 0.75mA current source is shown in Figure (3-15). The DC transfer characteristic of this source, Figure (3-16), illustrates that the bias current drawn is a function of the collector-emitter voltage (V_{CE}) at Q_{SOURCE}. More specifically, it is seen that V_{CE} must be greater than 0.3 volts in order to ensure that 0.75mA is drawn. This represents a critical design parameter for establishing a proper DC bias on the current source.
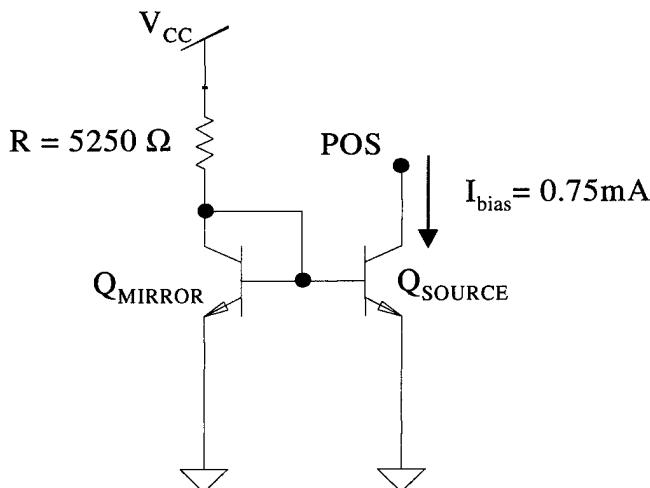
Figure 3-15.   Final Design of a Practical 0.75mA Current Source.

The 0.75mA current source design is validated by a direct performance comparison with an ideal current source. Figure (3-17) compares the output signals for a maximally loaded inverter/buffer circuit when driven by both
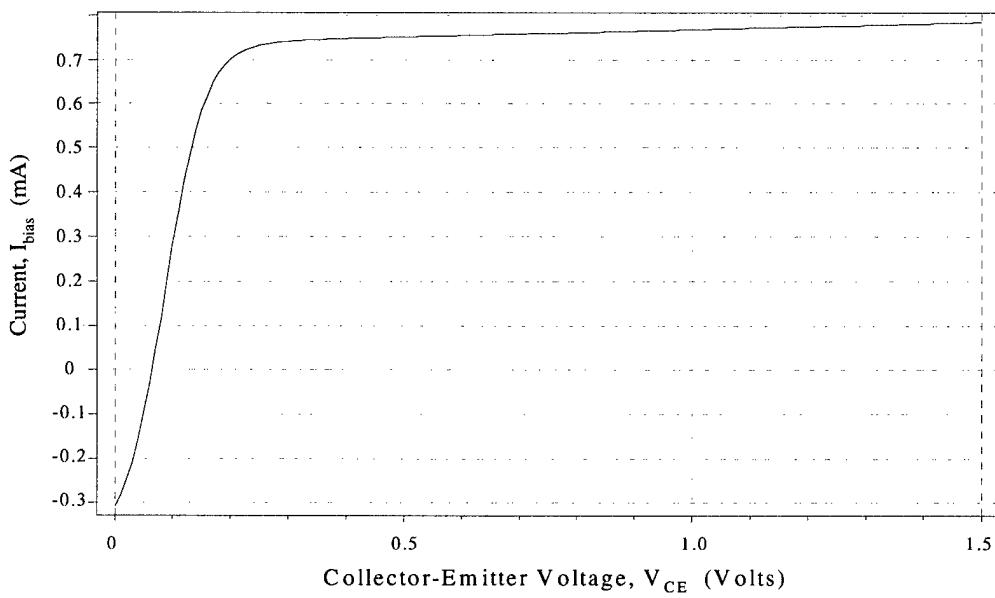
Figure 3-16. **Transfer Characteristic of the 0.75mA Current Source.**
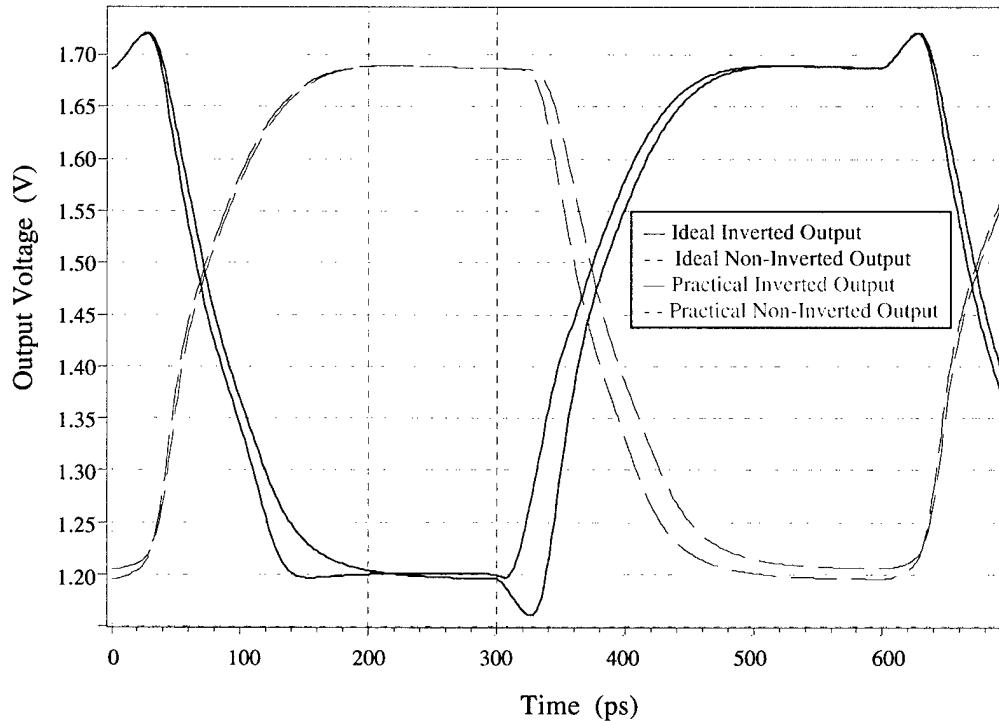
Figure 3-17. **Comparison of Inverter Performance, Practical Current Source vs. an Ideal Source.**

67

an ideal and a practical current source. It can be seen that the transition delay resulting from the practical source is consistently ahead of the ideal source for the inverted output signal by a margin of five picoseconds. Meanwhile, the non-inverted output signal of the practical current source maintains the status quo by matching the pair delay of the ideal source. In a design that is characterized by alternating stages of positive and negative logic signals, it is reasonable to expect that the implementation of the practical current source would yield a slight improvement over the ideal source.

b) 2.0mA Current Source

Exercising a little foresight into the conclusions of Chapter IV, it is convenient here to present the design of the 2mA practical current source. This design is a simple modification to the 0.75mA design — implemented by decreasing the resistance from 5250 Ω to 2020 Ω. This allows an increase of current flow into the base of Q_{MIRROR} and produces the transfer characteristic shown in Figure (3-18). Again, a bias voltage at Q_{MIRROR} must ensure that V_{CE} is greater than or equal to 0.3 volts in order to achieve proper functioning of the current source.

The 2mA current source is also validated by testing it against an ideal current source while driving a maximally loaded D-type CML Latch. The respective output
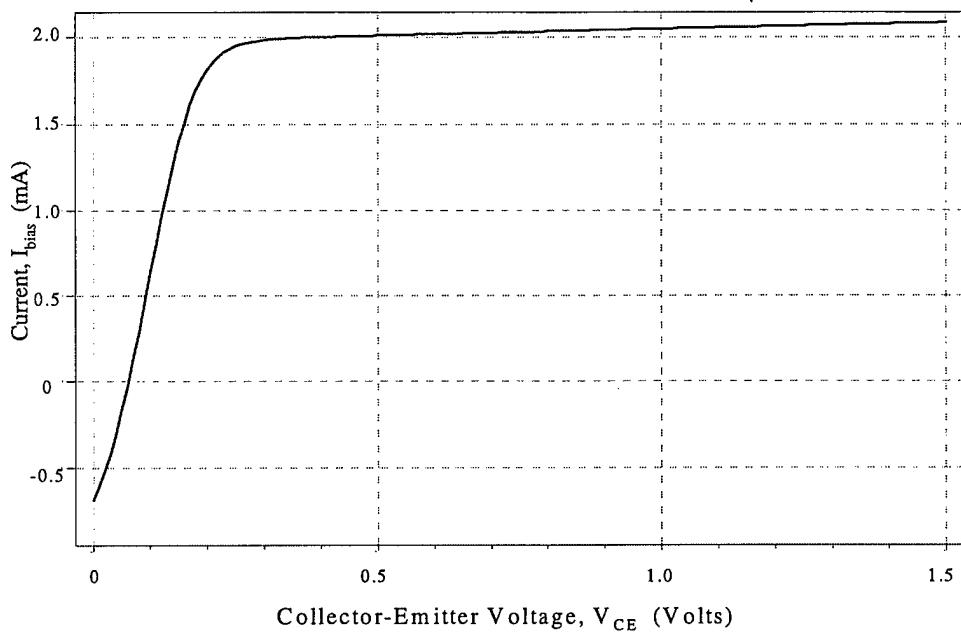
Figure 3-18. Transfer Characteristic of the 2.0mA Current Source.

signals, Q and QN, are plotted in Figure (3-19). It can be seen that the output signal transition delay resulting from the practical source compares favorably with the delay associated with the ideal source. However, the ideal-driven output signals consistently crosses the reference voltage of 1.45 volts approximately 10 picoseconds ahead of the practical-source-driven output signals. Thus, the effective margin of error for approximating the practical source with an ideal source is 10 picoseconds. In a synchronous pipelined architecture, this simply adds between 10 and 20 picoseconds to the minimum clock period.
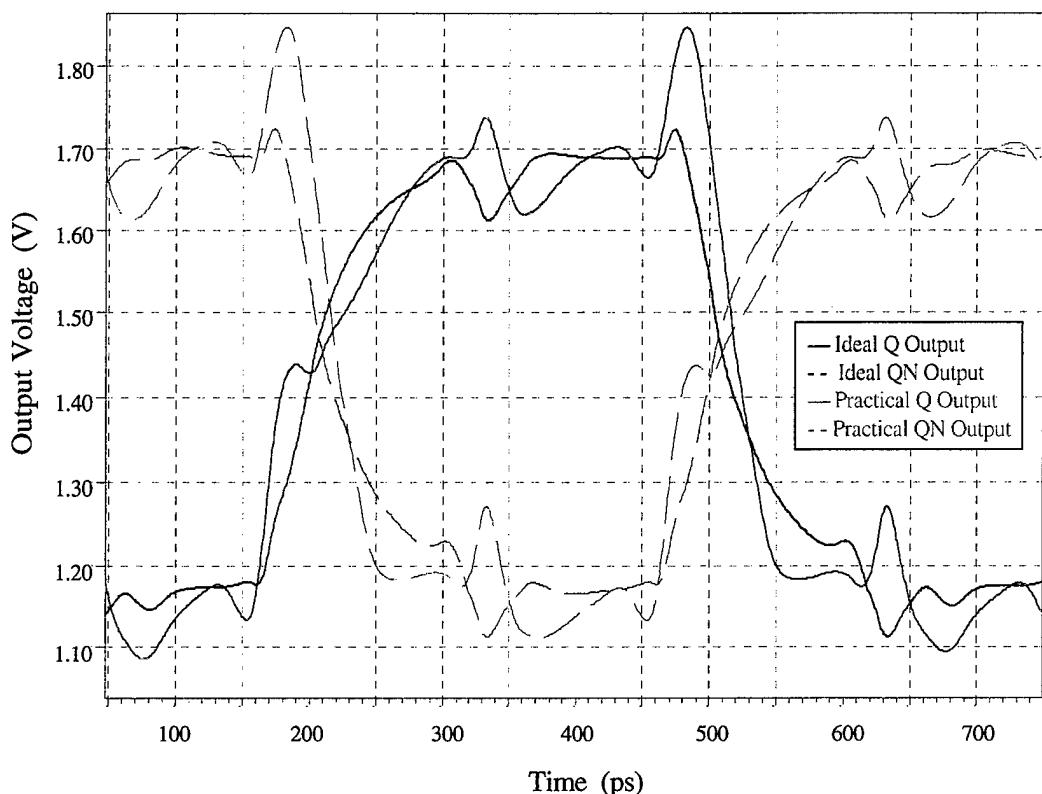
**Figure 3-19. Comparison of Latch Performance, Practical
Current Source vs. an Ideal Source.**

In summary, a sufficient set of logic circuits is now
in hand, along with a practical current source with which to
drive them. Thus, the combinational logic for a multiplier
can be fully implemented. However, based upon the intent of
pipelining this multiplier, it is necessary to construct the
clock-driven devices that will control the flow of data.
Chapter IV presents this discussion with the design of a D-
type latch, a D-type flip-flop, and a clock driver.

IV. HBT CML LATCH AND REGISTER DESIGN

A. LATCH DESIGN

1. Circuit Topology

a) *Two Latch Topologies*

The most common latch design is based upon the logic level schematic illustrated in Figure (4-1). Design of this latch simply requires the proper connection of four NOR gates with the appropriate clock and logic input signals. The cumulative power consumed by the four NOR gates constitutes a significant cost (based upon the four milliwatt per gate design from Chapter III).
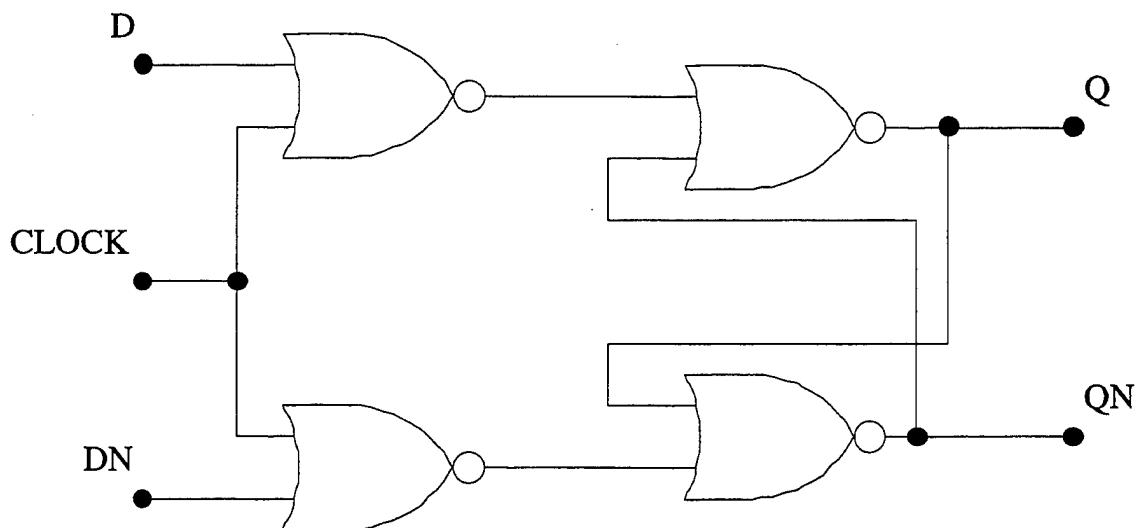
Figure 4-1. D-type Latch constructed from NOR gates.

However, the unique characteristics of CML provide an alternative design that yields comparable performance at a significant savings in power. This CML latch design is illustrated in Figure (4-2). Due to the relative unfamiliarity of this design, a brief functional description follows.
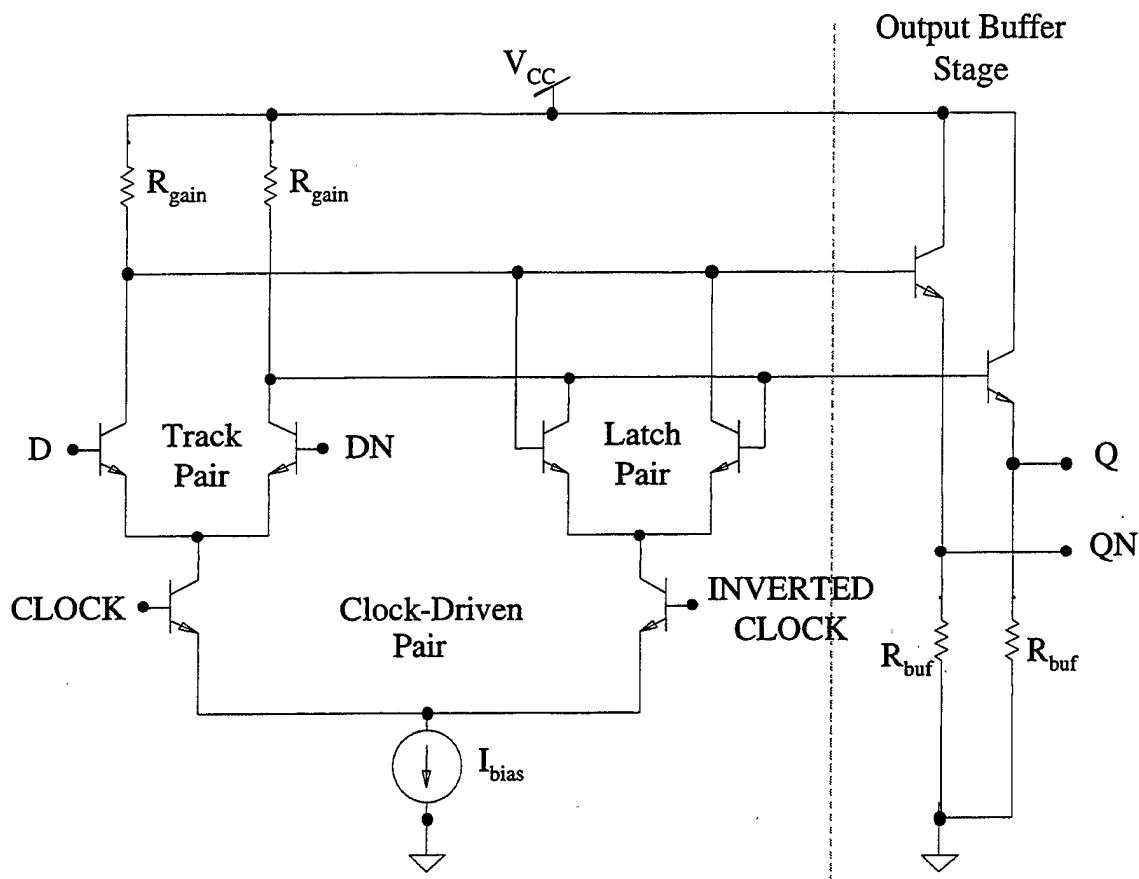
Figure 4-2. CML D-type Latch Design (After Jalali).

b) *Functional Description of a CML Latch*

Referencing Figure (4-2), the source labeled I_{bias} draws a constant current through the lower (clock-driven) differential pair. Complementary clock signals provide the differential inputs. Depending upon the phase of the clock signal, current is drawn from one of the two cascaded differential pairs, i.e. either the track pair or the latch pair. Consider the case when the CLK signal is *high*. Current will be drawn from the "track" pair while the "latch" pair is simultaneously cut off. In this case the latch is considered "open" or "transparent," and the track pair behaves like the differential pair configuration of the inverter/buffer logic gate. Thus, the logic inputs of the track pair are mirrored at the opposite collector. However, there is one exception. In the CML latch, complementary logic inputs are employed rather than a logic reference voltage. For a single logic input, complementary input signals enhance noise immunity and provide for symmetric waveforms at the complementary output ports.

Now, consider when the CLK signal transitions from *high* to *low*. The track pair is cutoff as current is switched to the latch pair via the right side of the clock-driven differential pair. Herein lies the significance of the common collector nodes shared by the track pair and latch pair. Due to the high impedance nature of the HBT

73

collector-base junction, the voltage level at the collector is slow to change and lingers long enough to bias the latch pair for essentially identical operation and output levels. This effectively latches the logic levels from the track pair to the latch pair. (Jalali, 1995)

Regardless of the state of the latch, the logic levels at the common collector (of the track and latch pairs) are reflected at the latch output ports via the same output buffer configuration presented in Chapter III.

2.   Initial Conditions and Design Parameters

The CML latch presents the most demanding DC bias requirements of any circuit designed for this project. As a result, no voltage cap has been placed upon its design. Rather, the initial design goal is to determine the minimum necessary DC bias conditions for proper operation of the latch. The resulting "voltage budget" will define the voltage relationships for proper operation of each transistor and differential pair. It will further establish important specifications for supply voltage and logic signal levels. Derivation of the "voltage budget" is presented as part of the DC analysis in the following section.

The minimum available transistor area (1x1 micron) is employed for optimum switching speeds, and the fanout requirement remains at four. These specifications are

74

consistent with the logic circuits designed in the previous
chapter.

3. DC Analysis

a) DC Bias Conditions / The Voltage Budget

For proper operation of the CML latch, each
differential pair of transistors must be properly biased.
Knowing the requirements imposed by proper DC bias
conditions will reveal the following necessary design
parameters:

- Required minimum supply voltage

- Required minimum voltage level for
  representing the positive (*high*) phase of
  the clock

- Required minimum voltage level for
  representing a logic *high* state

- Maximum allowable signal range between
  high and *low* logic levels

To facilitate analysis, the CML latch topology is divided
into three levels of operation, as illustrated in Figure
(4-3). Level one (the bottom level) is a practical current
source. Implementing the design from Chapter III-E, the
current source requires a minimum of V_{Ibias} volts at node X in
order to sustain the desired level of bias current.

(4-1) $$V_X \geq V_{Ibias}$$

$$V_{C(low)} \geq V_{LOGIC(hi)} - V_{BE(on)|Q34} + V_{CE(sat)}$$

$$V_{C(low)} \geq V_{C(hi)} - V_{BE(on)|Q56} + V_{CE(sat)}$$

$$V_{RANGE} \leq V_{BE(on)|Q56} - V_{CE(sat)}$$

$$V_{LOGIC(hi)} - V_{RANGE} = V_{C(low)} - V_{BE(on)|buffer}$$

$$V_{LOGIC(low)} + V_{RANGE} = V_{C(hi)} - V_{BE(on)|buffer}$$

$$V_{LOGIC(hi)} \geq V_{BE(on)|Q34} + V_{Y1}$$

$$V_Y \geq V_{CLK(hi)} - V_{BE(on)|Q12} + V_{CE(sat)}$$

$$V_{CLK(hi)} \geq V_X + V_{BE(on)|Q12}$$

$$V_X \geq V_{Ibias}$$
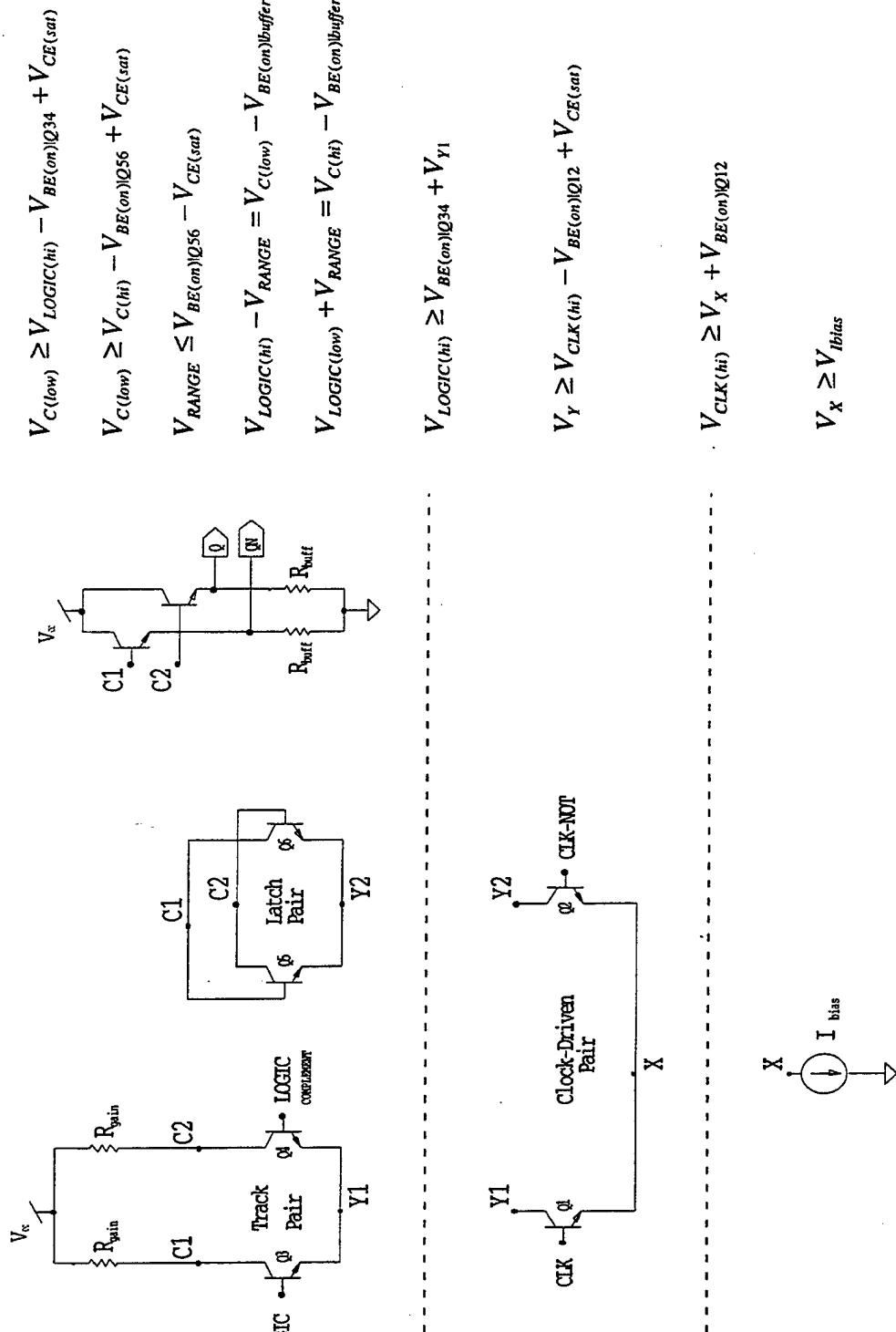
Figure 4-3. Voltage Budget for the CML Latch

76

This requirement imposes the following operational condition upon the "driving" base voltage of the Q_1/Q_2 differential pair (i.e. the high CLK voltage).

(4-2) $$V_{CLK(hi)} \geq V_X + V_{BE(on)|Q12}$$

A further consideration is the proper biasing of the Q_1/Q_2 collectors for operation in the active region. This places the following operational condition upon the collector voltages (nodes Y1 and Y2).

(4-3) $$V_Y \geq V_{CLK(hi)} - V_{BE(on)|Q12} + V_{CE(sat)}$$

where, V_Y represents either V_{Y1} or V_{Y2} Only the tracking differential pair (connected to node Y1) will be addressed at this point because it is driven by lower voltage levels which impose more restrictive DC bias conditions on Y1 than Y2.

Once again, a minimum voltage requirement at the common emitter of the Q_3/Q_4 differential pair presents a constraint on the minimum steady-state driving voltage at each base. This driving voltage corresponds to a logic high input voltage. Thus, the voltage level selected to represent a logic *high* must satisfy the following relationship.

(4-4) $$V_{LOGIC(hi)} \geq V_{BE(on)|Q34} + V_{Y1}$$

Finally, three conditions must be satisfied at the collectors of the track pair. The first condition is that

77

transistors Q_3 and Q_4 must operate in the active mode. This requires the following familiar relationship.

(4-5) $$V_{C(low)} \geq V_{LOGIC(hi)} - V_{BE(on)|Q34} + V_{CE(sat)}$$

where V_c represents either V_{c1} or V_{c2}

Similarly, the second condition requires that the transistors of the latch pair also operate in the active mode. This condition differs from the one above because the latch pair is driven by the collector voltage levels of the track pair.

(4-6) $$V_{C(low)} \geq V_{C(hi)} - V_{BE(on)|Q56} + V_{CE(sat)}$$

Defining the voltage range of the logic signal (V_{RANGE}) as the difference between high and low voltage levels, Equation (4-5) is manipulated to show the maximum value.

(4-7) $$V_{RANGE} \leq V_{BE(on)|Q56} - V_{CE(sat)}$$

Knowing the transistor parameters for $V_{BE(on)}$ and $V_{CE(sat)}$ from Chapter II, $(V_{RANGE})_{max}$ is 0.5 volts.

The third condition is that the input and output logic levels must match. A high logic input ($V_{LOGIC(hi)}$) at the transistor base must drive the collector voltage relatively low ($V_{c(low)}$) such that it produces a matched low logic output at QN. Likewise, the inverse must also be true. The following equations express these requirements.

(4-8) $$V_{LOGIC(hi)} - V_{RANGE} = V_{C(low)} - V_{BE(on)|buffer}$$

(4-9) $$V_{LOGIC(low)} + V_{RANGE} = V_{C(hi)} - V_{BE(on)|buffer}$$

Based upon these relationships the maximum collector voltage is determined, which further dictates the minimum required supply voltage for proper DC operating conditions.

The voltage budget relationships are summarized in Figure (4-3). Actual values have been determined for four latch configurations as listed in Table (4-1). The essential difference is the magnitude of the bias current. An economical margin of safety has been built into these values.

Notice that these margins have been allowed to vary slightly between configurations in order to maintain uniform values for clock and logic signal values. This greatly simplifies the comparative testing of the four configurations. The design margins are highlighted to illustrate the negligible deviation incurred. All four configurations meet and exceed the required DC bias conditions. In the event that uniform design margins had been used such that the supply voltages were optimized, the difference would have been trivial — within plus or minus 0.1 volt or 4% of the 2.5 volt supply voltage.

b) DC Bias Optimization

At this point the gain resistance, buffer resistance, and the bias current are the only undetermined parameters. The same procedures described in the design of the inverter/buffer circuit are employed to design four

CML Latch Voltage Budget
for Multiple Bias Current Configurations

	1mA	1.5mA	2mA	3mA
Known/Measured Parameters:				
$V_{BE(on)}$	0.775	0.80	0.82	0.857
$V_{CE(sat)}$	0.26	0.30	0.31	0.35
$V_{I\text{-}bias}$	0.3	0.3	0.3	0.3
Determined Parameters:				
$[V_{RANGE}]_{max}$	0.515	0.5	0.51	0.507
Margin for Range of Logic Signal Voltage	0.015	0.0	0.1	0.007
$[V_{RANGE}]_{actual}$	0.5	0.5	0.5	0.5
V_{CC}	2.5	2.5	2.5	2.5
Margin to nearest tenth of a volt V_{CC}	0.075	0.025	0.025	0.0
$V_{C(hi)}$	2.425	2.475	2.475	2.5
$[V_{LOGIC(hi)}]_{actual}$	1.7	1.7	1.7	1.7
Margin for Differential Logic Signal Switching	0.24	0.2	0.19	0.15
$[V_{LOGIC(hi)}]_{min}$	1.46	1.5	1.51	1.55
V_{Y1}	0.685	0.7	0.69	0.693
$V_{CLK(hi)}$	1.2	1.2	1.2	1.2
V_X	0.42	0.4	0.39	0.358
Margin for Differential Clock Signal Switching	0.12	0.1	0.09	0.058
$V_{I\text{-}bias}$	0.3	0.3	0.3	0.3
Based upon a 0.5 volt signal swing for both logic and clock signals:				
$V_{LOGIC(low)}$	1.2	1.2	1.2	1.2
$V_{CLK(low)}$	0.7	0.7	0.7	0.7

Table 4-1. Voltage Budget for the CML D-type Latch.

different latch configurations based upon the specifications determined in Table (4-1).

Noise Margins are obtained from the DC transfer characteristic of each. These results are included in Table (4-2). With maximum fanout loads on both output ports, all four CML latch designs meet the requirements of a 0.5 volt output signal range and 0.1 volt (20%) balanced noise margins. Therefore, all four CML designs are considered in transient analysis.

Bias Current (mA)	Gain Resistor (Ohms)	Buffer Resistor (Ohms)	No Load / Loaded High Noise Margin (Volts)	No Load / Loaded Low Noise Margin (Volts)	Logic Signal Range (Volts)
1	600	2000	0.14 / 0.13	0.13 / 0.13	0.49
1.5	410	2000	0.13 / 0.13	0.13 / 0.13	0.51
2	310	2000	0.12 / 0.12	0.12 / 0.12	0.51
3	210	2000	0.11 / 0.11	0.11 / 0.11	0.52

Table 4-2. Results of DC Analysis.

4. AC/Transient Analysis

a) Performance Parameters

Three parameters are of primary interest in evaluating the transient performance of a latch: setup time, hold time, and logic propagation delay. Figure (4-4) illustrates how each of these relates to the events on a transient plot. In the absence of a reference voltage,

81
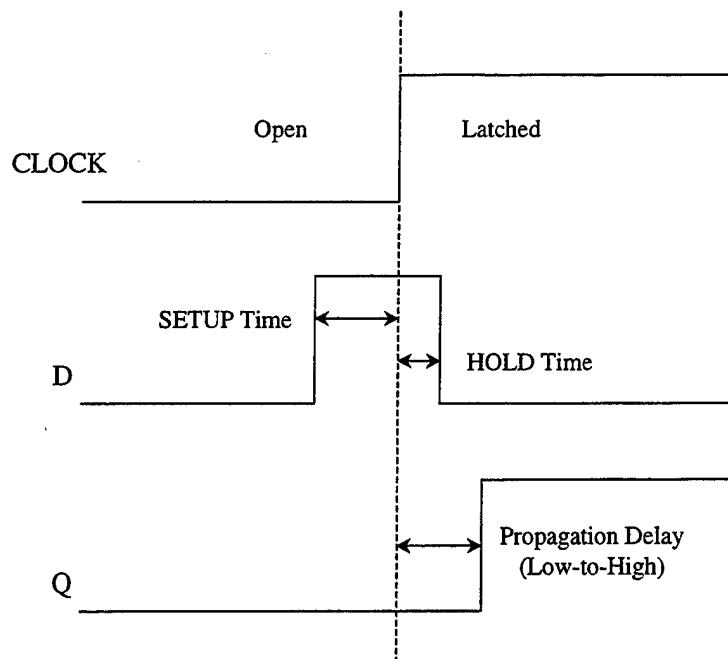
Figure 4-4. Illustration of setup time, hold time, and propagation delay.

differential signal references are taken as the point where the complementary signals cross.

As a figure of merit for optimizing the trade-off between speed and power, a power-delay product is calculated using the values defined here. The figure for power represents the average power, and the figure for delay represents the sum of the setup time and the worst-case propagation delay time.

b) Analysis Procedures

For an accurate evaluation of latch performance, it is necessary to provide realistic logic and clock input signals as well as realistic worst-case fanout loads.

Furthermore, to ensure and demonstrate the proper DC bias design of the CML latch, practical current sources are implemented in testing.

In addition to the four CML latch designs, the traditional logic latch is also tested. Each design is substituted into the test circuit to determine the performance parameters described in the previous section.

c) *Summary of Results*

The results of transient analysis are summarized in Table (4-3). The 1.5mA configuration achieves the minimum power-delay product as illustrated in Figure (4-5). Note, however, that the 2mA configuration performs at a
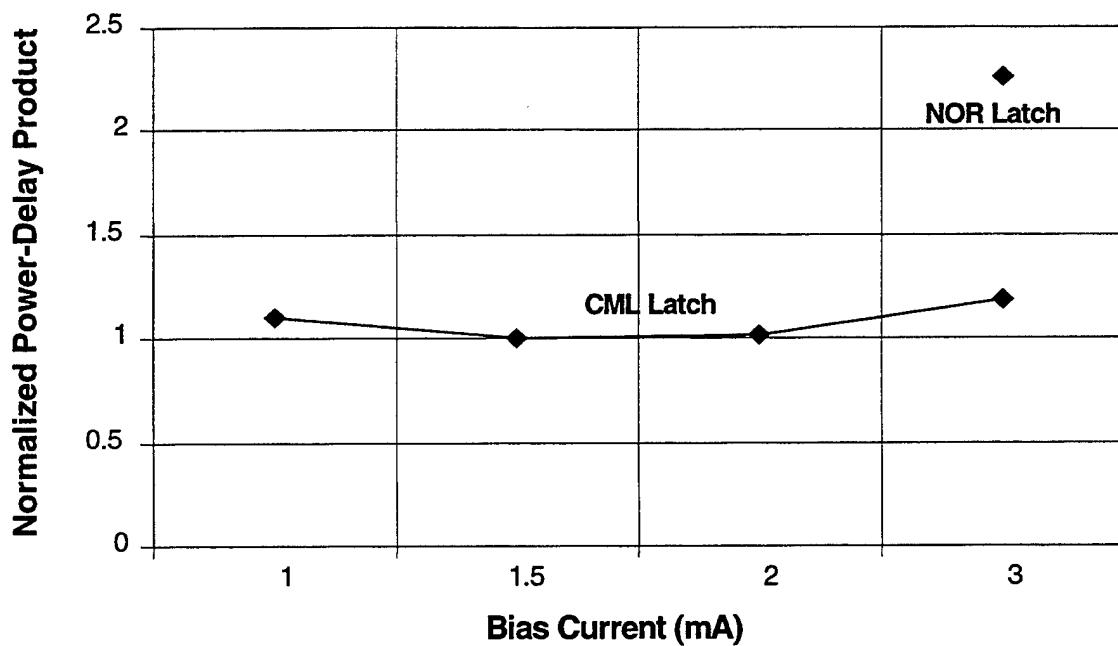
Figure 4-5. Results of Transient Analysis: Normalized Power-Delay Product of Latch Configurations.

83

Latch Topology	Bias Current (mA)	Current per Latch (mA)	t_{setup} Setup Time (pS)	t_{hold} Hold Time (pS)	t_{prop} H-L (pS)	t_{prop} L-H (pS)	Max Total Delay (pS)	Power-Delay Product (mW-pS)	Normalized Power-Delay Product
CML Latch	1	2.6	52	13	40	7.5	92	593	1.10
	1.5	3.1	40	14	31	5	71	541	**1.00**
	2	3.6	35	13	27	4.5	62	550	**1.02**
	3	4.5	25	18	23	32	57	641	1.18
NOR Latch	3	6.4	33.5	44	4	30	107.5	1184	2.26

Table 4-3. Results of Transient Analysis for Latch Configurations.

comparable level of efficiency.   In the interest of
maximizing speed, it is a reasonable design trade-off to
sacrifice two percent efficiency in order to acquire a 12
percent reduction in latch delay.   Thus, the 2mA CML latch
configuration is selected for the implementation of a D-type
latch.

Regardless of the configuration, switching noise
proves to be a prominent characteristic of transient
performance in the CML latch.   Figure (4-6) illustrates the
effect of switching noise on the latch output, Q.   The noise
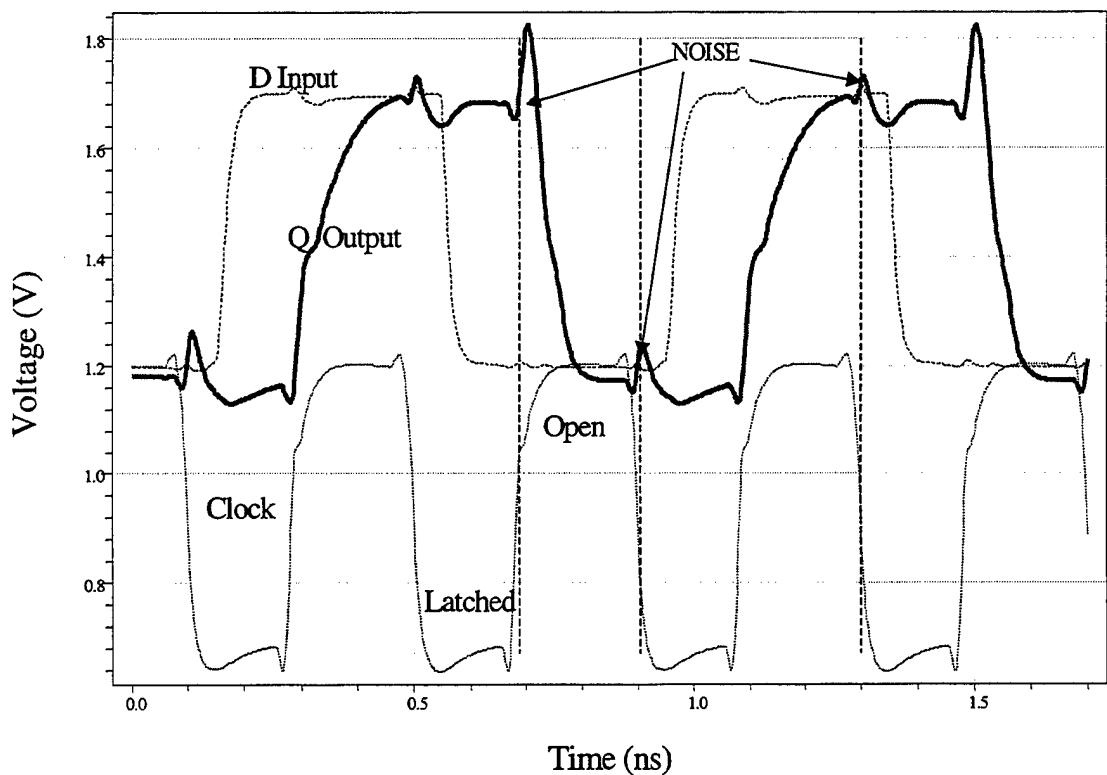
Figure 4-6.   Switching Noise in the CML Latch Output.

85

indicates a capacitive spike at the mutual collector nodes of the latch and track differential pairs. This results each time the clock-driven pair switches current to the opposite side. It is not expected that this noise will adversely affect the ability of the CML latch to drive reliable logic levels. However, in the event that the CML latch is overcome by noise, the NOR latch configuration is a viable alternative because it does not experience this problem.

Finally, the switching activity of the differential pair also induces variations in the current drawn from the supply voltage. Figure (4-7) illustrates these power rail transients for a single CML latch. The
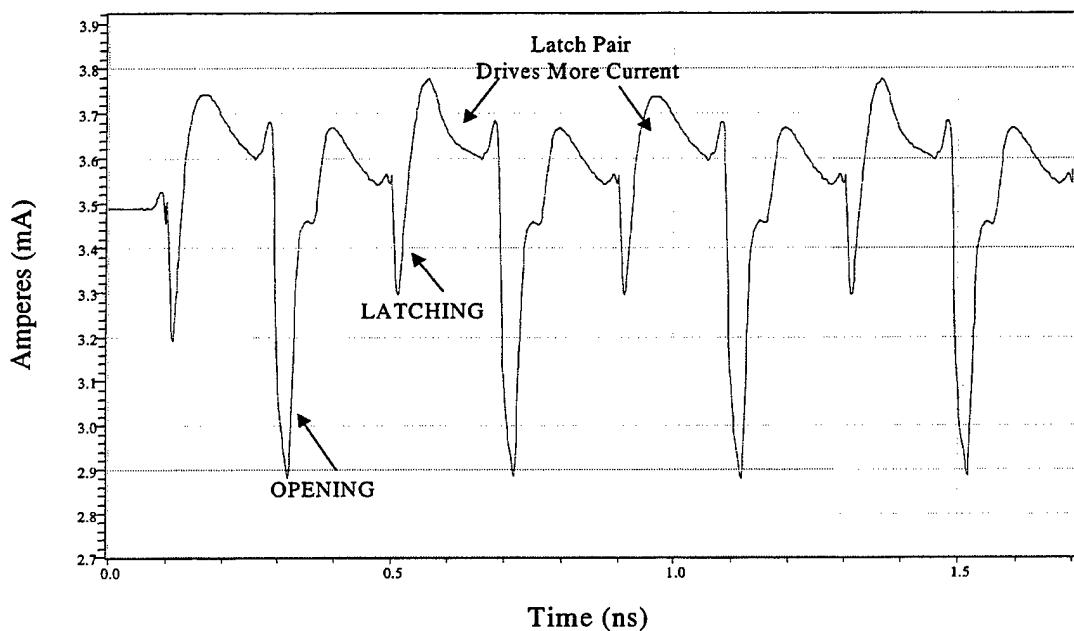
Figure 4-7. Power Rail transients due to the switching activity of a single CML Latch.

abrupt, periodic reduction in supply current coincides with the brief transition of current from one side of the differential pair to the other — driven by the switching of the clock signal. In the worst-case, this downward transient spike reaches a current level that is 18% below the average. It is also evident that slightly more current is drawn when the latch is *latched* because the latch pair is driven by a higher input voltage than the track pair. This results in a higher voltage and thus more current being drawn at the practical current source.

5. Special Latch Implementations

In the course of this design project, two special implementations of the CML latch have been designed. The first implements a logic reference voltage at one of the logic inputs of the latch. The purpose here is to eliminate the requirement for complementary logic signals at the multiplier input.

The second special implementation also uses a reference voltage; however, it does so with the purpose of conducting a logic function at the input to the latch. Although this circuit functions well, it actually results in slightly greater delays due to the increased collector capacitance at the tracking pair. As a result, it is not utilized in the multiplier circuit.

6. Final Design Summary: D-Latch

The final design for the CML latch is implemented with the parameters listed in Table (4-4) using the topology presented previously in Figure (4-2). Also listed are the transient performance parameters for operation at each level of fanout loading. These figures represent the performance of the latch when it is implemented with a practical current source and driven by a maximally loaded clock driver.

Latch
Design and Performance Summary

R_{gain}: 310 Ω

R_{buf}: 2000 Ω

I_{bias}: 2 mA

NM_L: 0.12 v

NM_H: 0.12 v

Power: 9.0 mW

Fanout Load (# gates)	Setup Time (pS)	Hold Time (pS)	t_{prop} H-L (pS)	t_{prop} L-H (pS)	Max Total Delay (pS)
1	33	9	27	0	60
2	33	10	28	1	61
3	34	10	31	2	65
4	35	10	34	3	69

Table 4-4. Final Design Summary of the D-type CML Latch.

B. FLIP-FLOP DESIGN (D-TYPE)

1. Overview and Analysis

The D-type flip-flop is constructed from two D-type CML latches. The two latches are connected in a master-slave configuration such that they are latched by opposite phases of the clock. This simple design is illustrated in Figure (4-7).
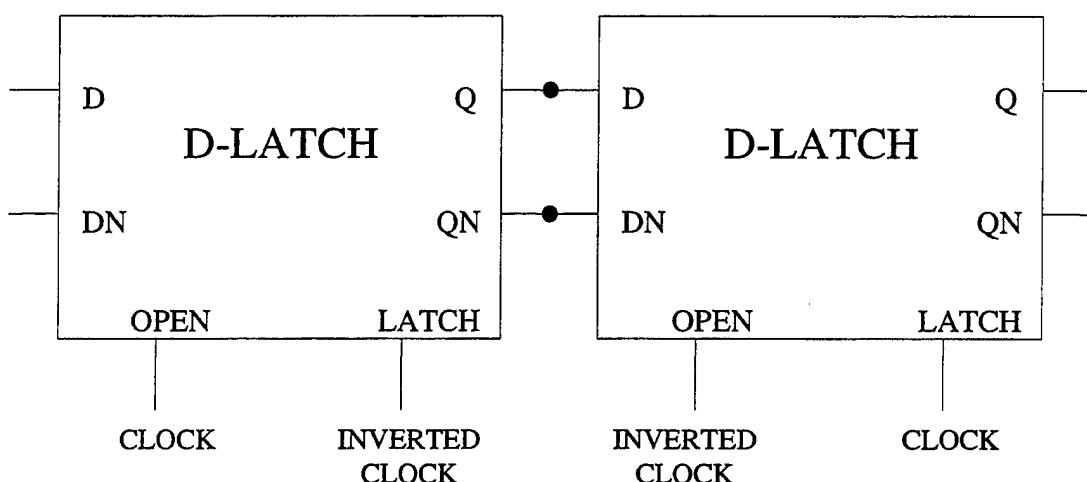
Figure 4-7. D-type Flip-Flop.

The flip-flop design is tested under the same conditions of loading and input signals as discussed previously for the latch. This testing verifies proper function of the flip-flop design and confirms that the flip-flop performance parameters of setup time and hold time mirror those of the CML latch. However, due to the presence of a second latch in the flip-flop, the propagation delays are greater.

2. Final Design Summary

The final design for the CML D-type flip-flop is essentially the master-slave configuration of two CML latches, as illustrated in Figure (4-7). The design parameters of the master and slave latches remains the same as shown in Table (4-4). The applicable performance parameters of the flip-flop have been summarized in Table (4-5).

Flip-Flop
Design and Performance Summary

Reference Latch Design Parameters
Power: 18 mW

Fanout Load (# gates)	Setup Time (pS)	Hold Time (pS)	t_{prop} H-L (pS)	t_{prop} L-H (pS)	Max Total Delay (pS)
1	33	9	49	35	**82**
2	33	9	53	47	**86**
3	34	9	52	45	**86**
4	35	10	54	43	**89**

Table 4-4. Design and Performance Summary of the D-type Flip-Flop.

C. CLOCK DRIVER DESIGN

1. Overview

The topology of the clock driver closely resembles that of the inverter/buffer circuit. In fact, the only necessary modification to the inverter/buffer design is a reduction of the output voltage range at the output buffer. This is accomplished by a simple voltage divider that effectively steps the voltage down to the desired voltage range between 0.7 and 1.2 volts (Figure 4-8). This voltage range is dictated by the CML latch design.

Two performance parameters are of particular interest in the clock driver design, fanout capability and the
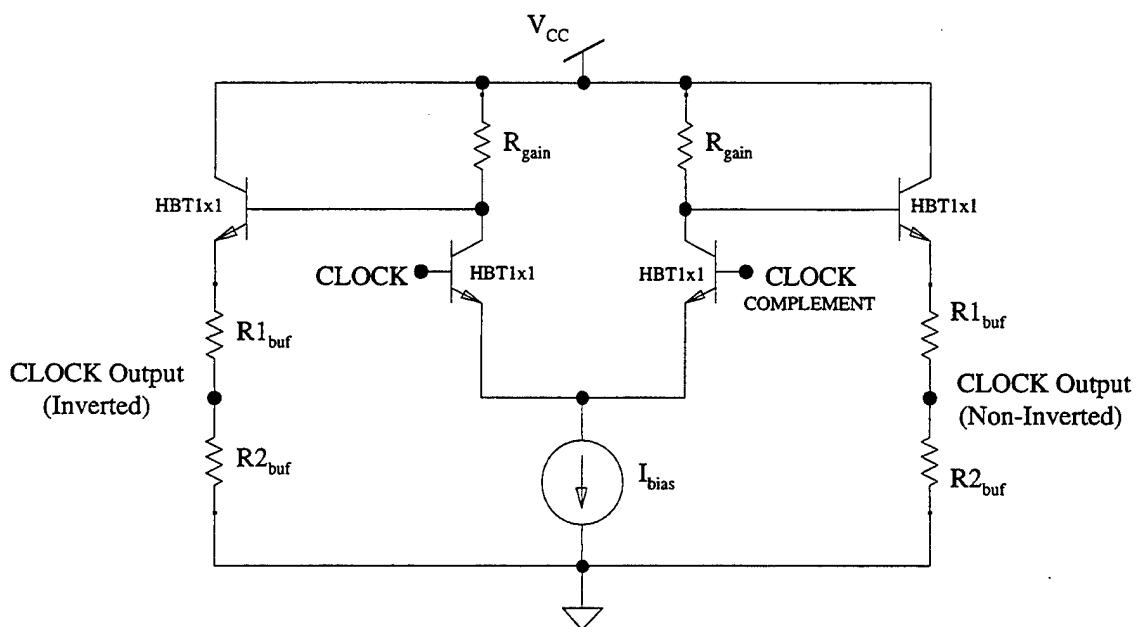
Figure 4-8. Topology of the Clock Driver Circuit.

symmetry of complementary output signals. Increased fanout is desirable to reduce the number of clock drivers required. Meanwhile, output symmetry is important to reduce clock skew between parallel clock paths. The absence of symmetry between the complementary output signals of the logic circuits (in Chapter III) results from the corresponding lack of symmetry between the input signals, i.e. the use of a reference voltage. Therefore, the clock driver is driven by the differential clock signals CLK and CLK-N.

2.   Analysis and Results

Fanout capability is maximized by the increase of current through the output buffer. Two further modifications to the inverter/buffer circuit make this possible. The first is to increase the bias current. For a supply voltage of 2.5 volts, a practical current source of 2mA is the largest that is operable without adversely biasing the circuit. Second, reducing the total resistance in the output buffer draws a larger base current and ultimately, more current is available to the output load.

For evaluation, the performance of two clock driver configurations is measured based upon the power consumed per load driven. The 1mA clock driver draws 5.5mA and consumes 13.8mW while driving a maximum of two latches. Meanwhile, the 2mA clock driver draws 6.5mA and consumes

16.3mW while driving four latches. Clearly, the 2mA clock driver is the desired implementation.

The synchronous switching behavior of the clock driver coupled with its high current consumption warrant an investigation of its power rail transient characteristic (Figure 4-9). It is not surprising that it follows the same periodic trend as discussed in the case of the CML latch. In the worst-case, the downward transient current spike deviates by 14.6% from the average current level. Also of
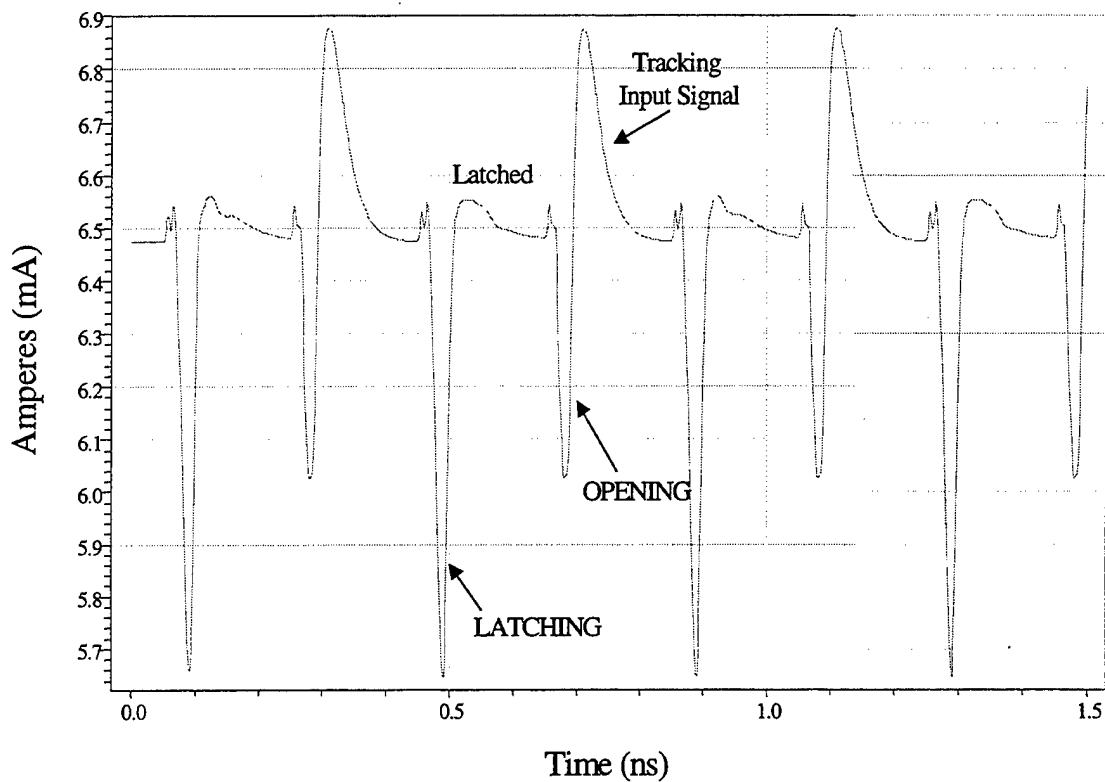
Figure 4-9. Power Rail transients induced by the switching activity of a single Clock Driver.

interest is the noise induced on the clocking signal by strong, simultaneous logic transitions at the latch input. As a result, a clock driver must be capable of driving a maximum fanout load of latches when the every latch input transitions simultaneously in the same direction.

3. Final Design Summary: Clock Driver

The final design for the clock driver is implemented with the parameters listed in Table (4-6) using the topology presented previously in Figure (4-8).

Clock Driver
Design and Performance Summary

R_{gain}:	400 Ω
$R1_{buf}$:	110 Ω
$R2_{buf}$:	450 Ω
I_{bias}:	2 mA
NM_L:	0.08 v
NM_H:	0.10 v
Power:	16.3 mW
Fanout:	4 Latches

Table 4-6. Design and Performance Summary of the Clock Driver Circuit.

At this point, the set of building blocks is complete. The logic circuits of Chapter III and the clock-driven devices of Chapter IV are brought together in Chapter V to implement several pipelined multiplier configurations.

V.   HBT CML PIPELINED MULTIPLIER DESIGN

A.   LOGIC STAGE DESIGN

1.   Overview

As introduced in Chapter II-C, the multiplier logic for this project is implemented with the three functional processes illustrated in Figure (5-1):   partial product generation,   carry-save   addition,   and   carry   completion
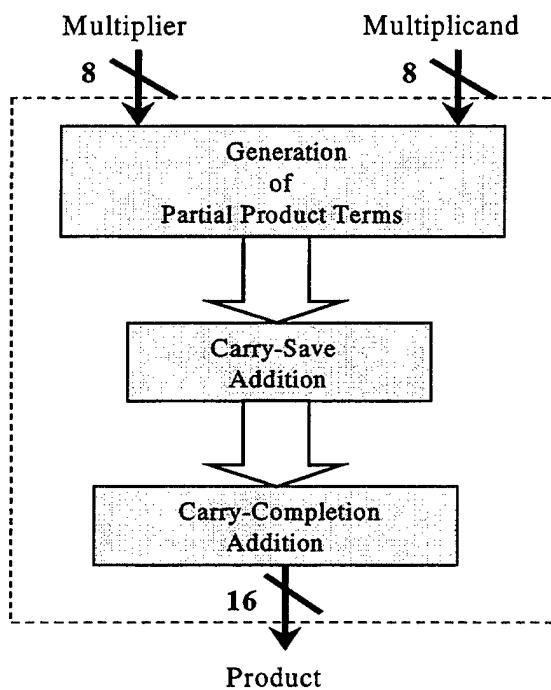
Figure 5-1.   Generalized Block Diagram of an 8x8 bit Multitplier.

addition.   In the case of the 8x8 bit multiplier which is implemented in this chapter, the process of carry-save addition is actually accomplished with successive stages of

carry-save adders. More specifically, the use of three-to-two carry-save adders produces the logic implementation illustrated in Figure (5-2). The detailed process of carry-save-addition is addressed in the following section; however, this block diagram accurately represents the functional design of the multiplier and establishes a graphic reference for the follow-on discussion.

2. Carry-Save Adders

Each three-to-two carry-save adder takes three operands and produces two outputs, a sum and a carry. However, the carry-save adder implementations are not identical, due to a slightly different input configuration that exists for the first carry-save adder stage than for the follow-on stages. Referencing Figure (5-3), the first carry-save adder receives three non-aligned n-bit partial products. As a result, it generates $n+2$ sum bits and n carry bits. Meanwhile, the follow-on stages each receive an aligned input pair comprised of the carry and sum terms generated by the preceding stage. The third input is the next partial product term, and it is shifted by one bit. Thus, the sum is only $n+1$ bits and the carry is still n bits.

In the case of either carry-save adder, only the most significant n bits of the sum term are passed on to the next adder stage. The remaining least significant bit(s) represent the next most significant bit(s) of the final

Partial Product #8   Partial Product #7   Partial Product #6   Partial Product #5   Partial Product #4   Partial Product #3   Partial Product #2   Partial Product #1

x128   x64   x32   x16   x8   x4   x2

Carry Save Adder

x2   8   8 C   S

Carry Save Adder

x2   8   8 C   S

Carry Save Adder

x2   8   8 C   S

Carry Save Adder

x2   8   8 C   S

Carry Save Adder

x2   8   8 C   S

Carry Save Adder

x2   8   8 C   S

Carry-Completion Adder (8x8 bits)

C = 8 bit Carry Term
S = 8 bit Sum Term
P = 16 bit Final Product

9

P[15:7]   P[6]   P[5]   P[4]   P[3]   P[2]   P[1]   P[0]

Figure 5-2. Logic Implementation of an 8x8 bit Multiplier using six stages of Carry-Save-Adders and a Carry-Completion Adder.

Carry-Save Adder #1

	PP1$_7$	PP1$_6$	PP1$_5$	PP1$_4$	PP1$_3$	PP1$_2$	PP1$_1$	PP1$_0$
PP2$_7$	PP2$_6$	PP2$_5$	PP2$_4$	PP2$_3$	PP2$_2$	PP2$_1$	PP2$_0$	
PP3$_7$	PP3$_6$	PP3$_5$	PP3$_4$	PP3$_3$	PP3$_2$	PP3$_1$	PP3$_0$		

S_7 S_6 S_5 S_4 S_3 S_2 S_1 S_0 P_1 P_0

C_7 C_6 C_5 C_4 C_3 C_2 C_1 C_0

S[7:0]

C[7:0]
(x2)

Carry-Save Adder #2

S_7	S_6	S_5	S_4	S_3	S_2	S_1	S_0
C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
PP4$_7$	PP4$_6$	PP4$_5$	PP4$_4$	PP4$_3$	PP4$_2$	PP4$_1$	PP4$_0$

S_8 S_7 S_6 S_5 S_4 S_3 S_2 S_1 P_3

C_7 C_6 C_5 C_4 C_3 C_2 C_1 C_0
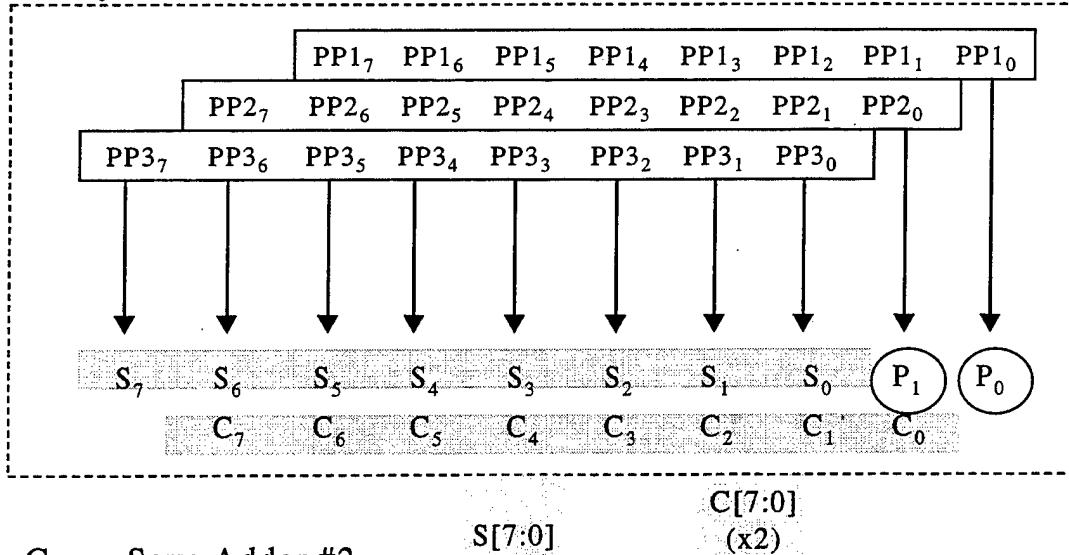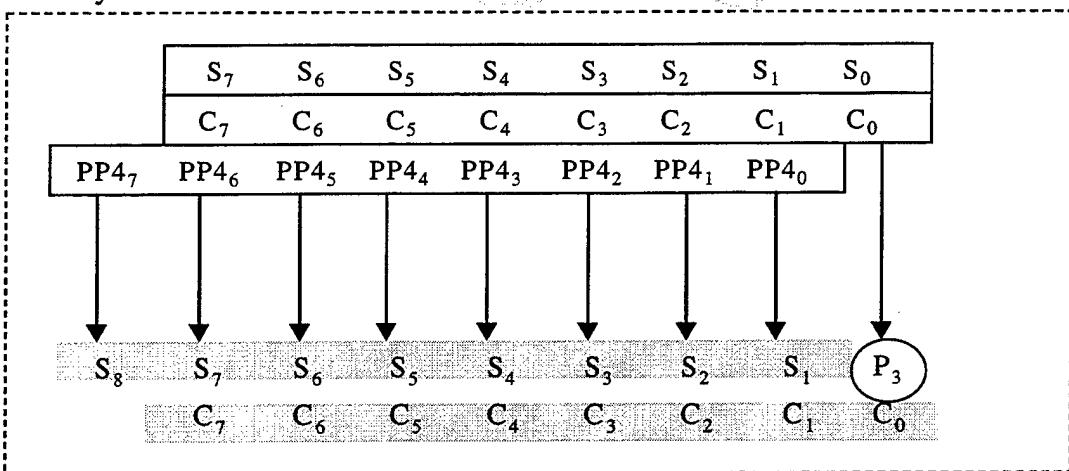
Figure 5-3. Functional Illustration of the two Carry-Save-Adder Implementations.

product and are passed directly to the multiplier output. These bits are highlighted with a circle in Figure (5-3). The final designs of the two carry-save-adder configurations are provided in Figures (5-4) and (5-5). Note the presence
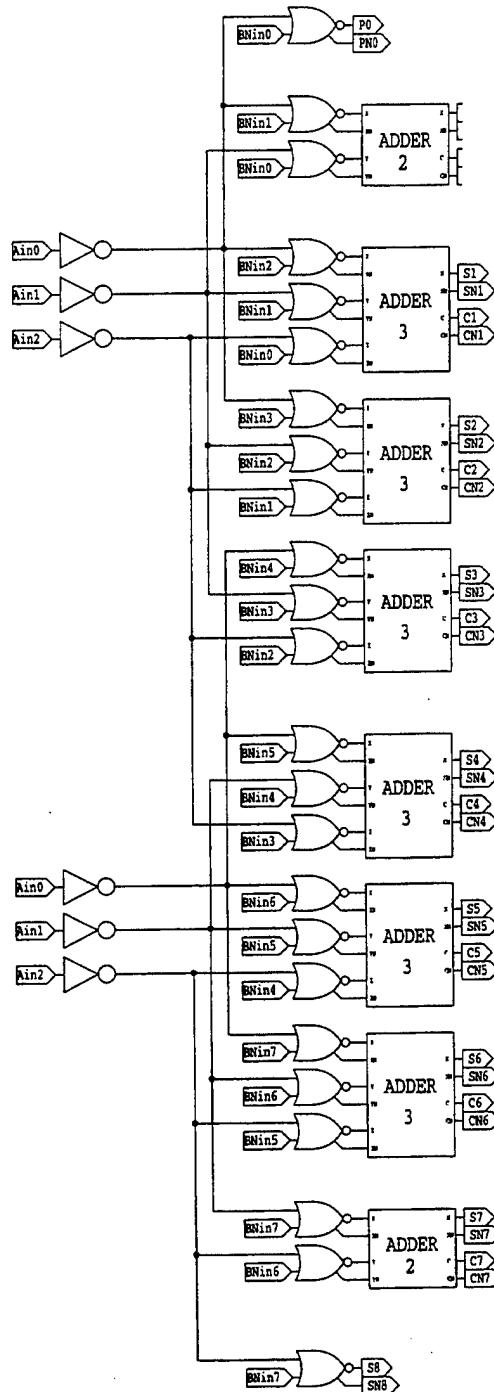
Input Signals

| Ain[2:0] | Multiplier Bits (3 for this stage) |
| BNin[7:0] | 8-bit Multiplicand |

Output Signals

P[1:0]	2 Product Bits
C[7:0]	8-bit Carry Term
S[8:1]	8-bit Sum Term

Figure 5-4.   Logic Schematic of Carry-Save-Adder #1.

99

Input Signals

Multiplier-Bit — Current Multiplier Bit for this stage

BNin[7:0] — 8-bit Multiplicand (Negated)

Output Signals

P[0] — Next Product Bit

C[7:0] — 8-bit Carry Term

S[8:1] — 8-bit Sum Term
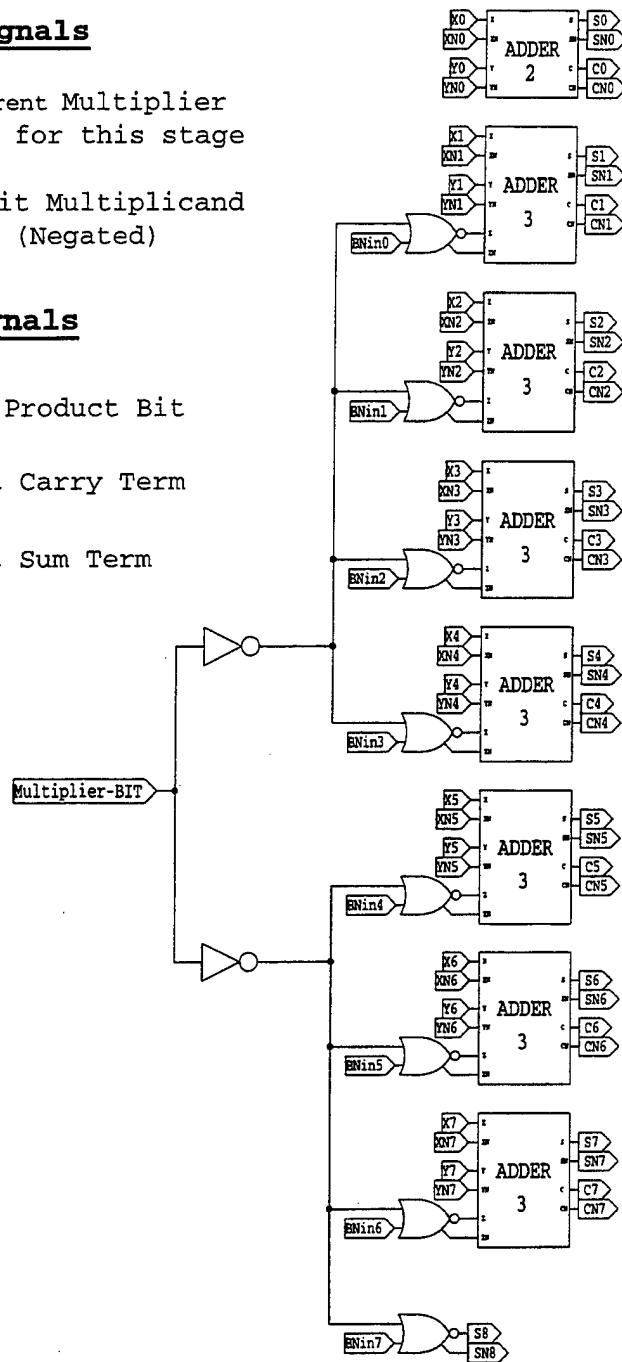
Figure 5-5. Logic Schematic of Carry-Save-Adder #2.

of more than simple adder circuits. A fanout limitation of four prevents a single signal from driving the eight input requirements for the current multiplier bit at each carry-save-adder stage. Thus, the arriving multiplier bits pass through an inverting buffer stage.

Furthermore, the OR/NOR gates are used to generate the partial product terms within each carry-save-adder stage, rather than at the multiplier input. Taking advantage of the complementary output signals available from the preceding register, the NOR gates perform a logical AND of each multiplicand bit with the appropriate multiplier bit. Local Generation of the partial product terms avoids the extensive requirement for intermediate registers that would be necessary to pass all partial product terms from one pipeline stage to the next (that is, referencing a scenario where all partial products are generated before the first carry-save adder).

3. Carry-Completion Adders

The carry-completion adder implements ripple-carry addition. This elementary design is preferred over carry-look-ahead addition because it facilitates a variety of simple pipeline implementations. Figure (5-6) illustrates the full carry-completion adder which can be conveniently segmented into as many as eight pipeline stages by separating the successive two and three-input adders.

101
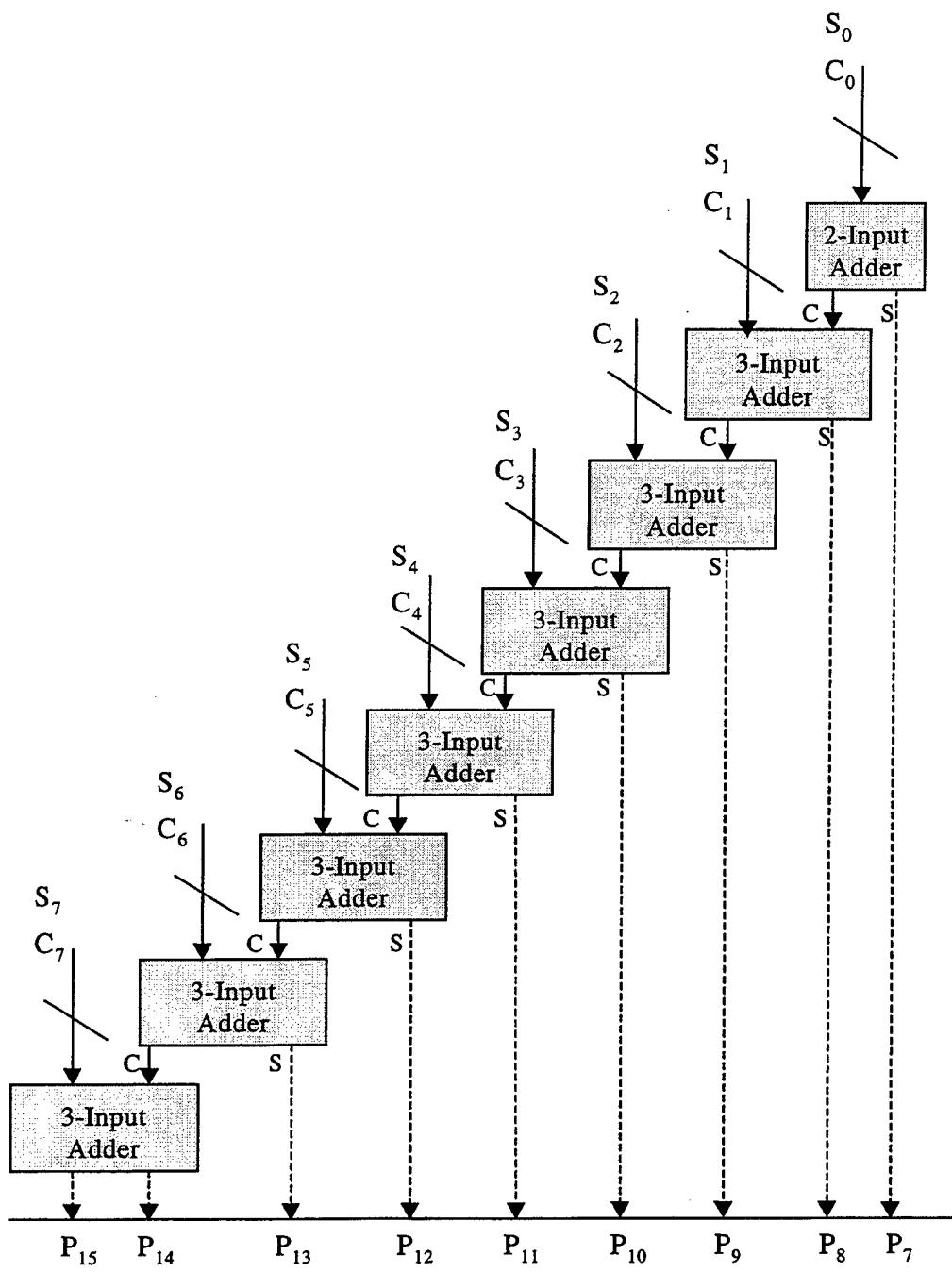
Figure 5-6. An 8-bit Ripple-Carry Adder to perform Carry-Completion.

B.   REGISTER STAGE DESIGN

Regardless of the number of pipeline stages, each multiplier implementation requires two eight-bit input registers and a sixteen-bit output register. For pipeline implementations with more than one stage, intermediate registers are also required. The size of these registers varies depending upon where the register is inserted in the flow of logic. All intermediate and output registers require complementary input signals. However, the input registers are distinctly designed to accept a single logic input signal for each bit, vice requiring complementary logic input signals. In order to accomplish this, the D-type flip-flops utilized in the input register must employ a special latch implementation which does not require differential input signals for the master latch of the master-slave flip-flop pair. The details of this latch implementation are presented in Chapter IV-A-5.

C.   CLOCK DISTRIBUTION

The purpose of the clock distribution scheme is to provide a local clock signal for clock-driven devices, namely the latches that comprise the registers described in the previous section. However, each clock driver can only sustain a maximum load of four latches, i.e., two flip-flops. Therefore, due to the number of clock-driven devices and the limited fanout capability of the clock drivers, the

103

clock signal must propagate through an extensive, multi-level distribution tree. As the number of clock-driven devices increases, the number of levels in this distribution tree must eventually increase as well. Thus, the more heavily pipelined multiplier implementations must make a larger investment of devices and power in clock distribution.

D.   MULTIPLIER IMPLEMENTATIONS

Five pipelined multiplier implementations have been designed for testing via Tanner SPICE simulation tools. These implementations include a one-stage pipeline, a two-stage pipeline, a four-stage pipeline, a six-stage pipeline, and a ten stage pipeline. The arithmetic logic is identical for each; however, the increased number of registers present in the more heavily pipelined implementations also implies a more extensive clock distribution tree. A block diagram of each implementation is presented in the following section.

E.   PERFORMANCE EVALUATION

1.   Evaluation Procedures

Prior to evaluation of the individual multiplier implementations, the multiplier logic is successfully tested with several operands in order to verify that it produces an accurate product. Following this verification, it is the goal of this performance evaluation to identify the maximum

operating clock frequency for each pipeline implementation. However, this can only be done once the critical path, i.e, the critical pipeline stage, is determined for each multiplier.

a) *Critical Path Identification*

The most direct and absolute means of identifying the critical path is to conduct full-length simulations of each multiplier for every possible combination and sequence of two 8-bit input operands. Conducting these nearly 4.3 billion simulations on each of the five multiplier designs is obviously prohibitive. Thus, the opposite extreme suggests that the worst-case transition delay be assumed for every logic circuit in every stage of the pipeline. While this successfully identifies an upper bound on the delay associated with the critical path, it is likely that the upper bound case does not exist as a result of two input operands. Furthermore, without knowledge of the input operands, simulations can not be conducted for verification.

Unfortunately, the logic behavior of the carry-save-adders makes an intuitive approach extremely difficult. Thus, a computer program designed by Kirk Shawhan, a research associate, has been utilized to identify the worst case input combinations. (Shawhan, 2000) The program effectively identifies a unique upper bound delay for each set of input operands. Those input combinations with the

105

worst-case upper-bound delays are then simulated to identify a single worst-case pair of operands and the critical stage where the most-delayed transition occurs. While it is not proven that this approach will identify the absolute critical path, it provides a reasonable and timely estimate for the purposes of this research.

b) Maximum Throughput/Clocking Frequency

Having determined the critical path, it is simply a matter of simulation time to identify the maximum clock frequency. For each pipeline implementation, a simulation is conducted which brackets the breakpoint of the multiplier. Furthermore, examination of the margin by which the setup time is met or missed provides a determination of the minimum clock period that is accurate within five picoseconds.

The increased number of devices in the more heavily pipelined designs made full-circuit simulation times extremely long. As a result, the breakpoints for the four-stage, the six-stage, and the ten-stage multipliers were determined from partial simulations. Only the critical stage and those stages immediately before and after it were simulated.

2. Performance Results of Each Implementation

The following ten pages provide a tw page design and performance summary for each of the five pipelined

multiplier implementations. Figure (5-7) illustrates the design and critical path of the one-stage multiplier on a block diagram. Table (5-1) provides a summary of data which quantifies circuit complexity, power consumption, data throughput rate and data latency of the one-stage pipelined multiplier. Finally, Figure (5-8) illustrates the success and failure of P14, the critical path, at clock frequencies below the above the breakpoint of the circuit.

Similarly, Figures (5-9) through (5-16) and Tables (5-2) through (5-5) provide the same performance results for the two, four, six, and ten-stage pipelined multipliers, respectively. A comparative analysis is conducted as a performance summary in the following section.

As a final note, all full multiplier simulations are conducted using ideal current sources. This decision saves numerous simulation hours without sacrificing valid transient performance data. A close correspondence has been demonstrated between the transient performance of the practical and ideal current sources for both the logic and the latch designs. Use of the ideal source, however, does produce overly optimistic power-consumption data due to the absence of power dissipation from the transistors in the practical current source. Therefore, the simulation data for current consumption is scaled to accurately represent the power consumed in practical implementation.

A = 1111 0111   B = 1100 0111

Critical Path Initiates
with the two operands
A=F7h, B=C7h

16-Bit Input Register

Carry Save Adder #1 (1)

Carry Save Adder #2 (2)

Carry Save Adder #2 (3)

Carry Save Adder #2 (4)

Carry Save Adder #2 (5)

Carry Save Adder #2 (6)

Signal Propagation

STAGE 1

Carry Completion Adder
(8 Bits)

Critical Path Terminates
with the LOW-to-HIGH
transition of P14

P15   P14

16-Bit Output Register

P = 1100 0000 0000 0001

Figure 5-7. One-stage pipelined multiplier implementation with an illustration of the critical path.

108

	Number of Transistors	Number of Resistors	Current (Amperes)	Power (Watts)
Logic	3952	2352	1.28	3.20
Registers	384	320	0.31	0.77
Clock	126	105	0.19	0.48
TOTAL	4462	2777	1.78	4.44

Maximum Throughput:	1.33	GHz
Latency:	0.75	Nano-second

Table 5-1.   Performance summary for the one-stage pipelined multiplier.
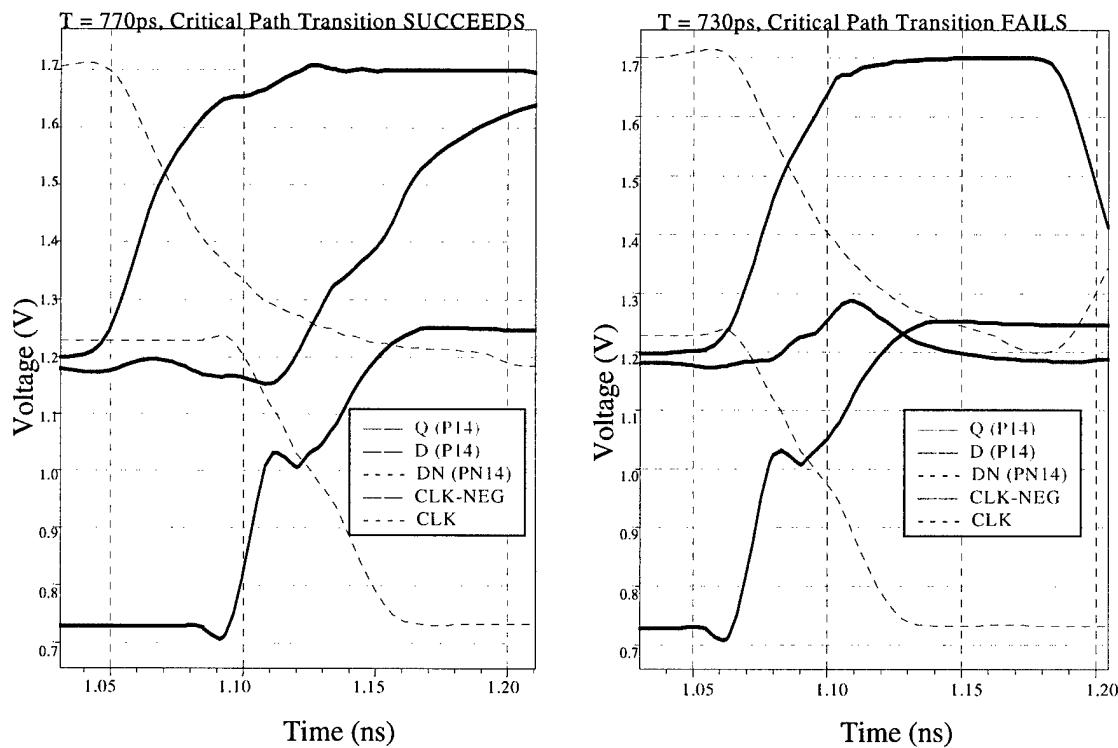
Figure 5-8.   Performance bracket of the minimum period for the one-stage pipeline multiplier.

A = 1111 0111   B = 1100 0111

Critical Path Initiates
with the two operands
A=F7h, B=C7h

16-Bit Input Register

Carry Save Adder #1 (1)

Carry Save Adder #2 (2)

Carry Save Adder #2 (3)

Carry Save Adder #2 (4)

Carry Save Adder #2 (5)

Carry Save Adder #2 (6)

STAGE 1

Signal Propagation

23-Bit Intermediate Register

Carry Completion Adder
(8 Bits)

STAGE 2

Critical Path Terminates
with the LOW-to-HIGH
transition of P14

P15 P14

16-Bit Output Register

P = 1100 0000 0000 0001

Figure 5-9. Two-stage pipelined multiplier implementation with an illustration of the critical path.

110

	Number of Transistors	Number of Resistors	Current (Amperes)	Power (Watts)
Logic	3952	2352	1.28	3.20
Registers	660	550	0.52	1.31
Clock	228	190	0.36	0.90
TOTAL	4840	3092	2.17	5.41

Maximum Throughput: 2.0 GHz

Latency: 1.0 Nano-second

**Table 5-2. Performance summary for the two-stage pipelined multiplier.**
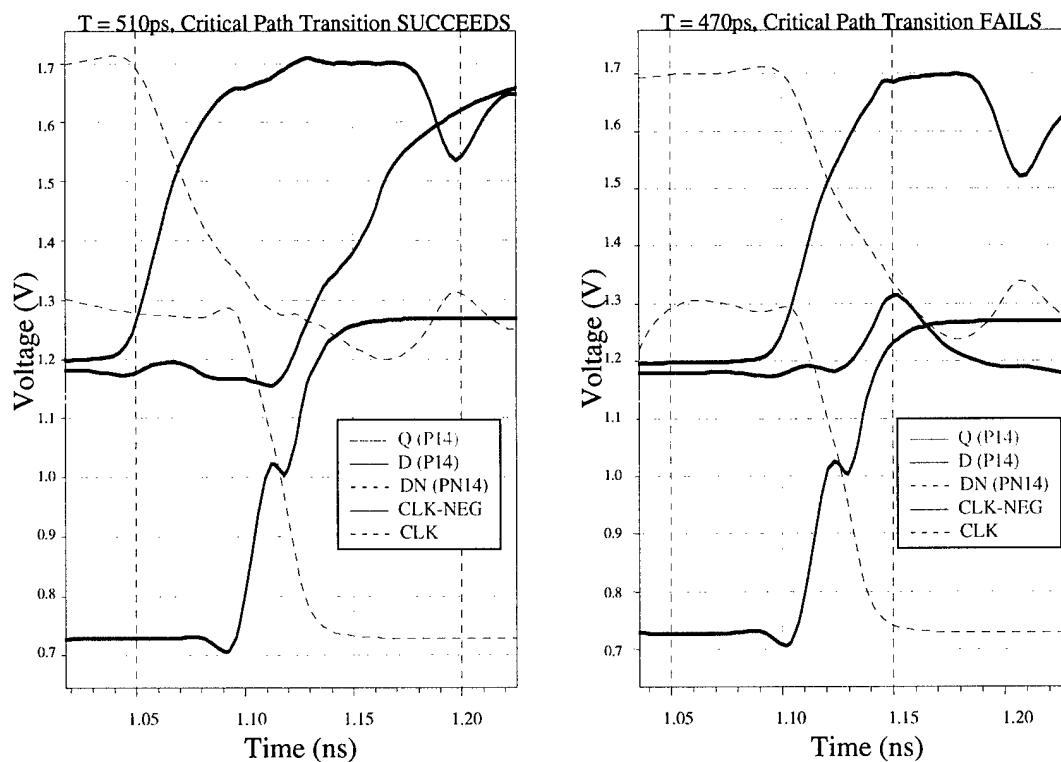
Figure 5-10. Performance bracket of the minimum period for the two-stage pipeline multiplier.

A = 1111 1111   B = 1000 0001

Critical Path Initiates
with the two operands
A=FFh, B=81h

16-Bit Input Register

Carry Save Adder #1 (1)

Carry Save Adder #2 (2)

Carry Save Adder #2 (3)

31-Bit Intermediate Register

Carry Save Adder #2 (4)

Carry Save Adder #2 (5)

Carry Save Adder #2 (6)

23-Bit Intermediate Register

Carry Completion Adder
(4 Bits)

20-Bit Intermediate Register

Signal Propagation

Critical Path Terminates
with the LOW-to-HIGH
transition of P15

Carry Completion Adder
(4 Bits)

P15 P14

16-Bit Output Register

P = 1000 0000 0111 1111

**Figure 5-11.   Four-stage pipelined multiplier
implementation with an illustration of the
critical path.**

112

	Number of Transistors	Number of Resistors	Current (Amperes)	Power (Watts)
Logic	3952	2352	1.28	3.20
Registers	1272	1060	1.01	2.52
Clock	438	365	0.68	1.71
TOTAL	5662	3777	2.97	7.43

Maximum Throughput:	3.45	GHz
Latency:	1.16	Nano-seconds

**Table 5-3.** **Performance summary for the four-stage pipelined multiplier.**
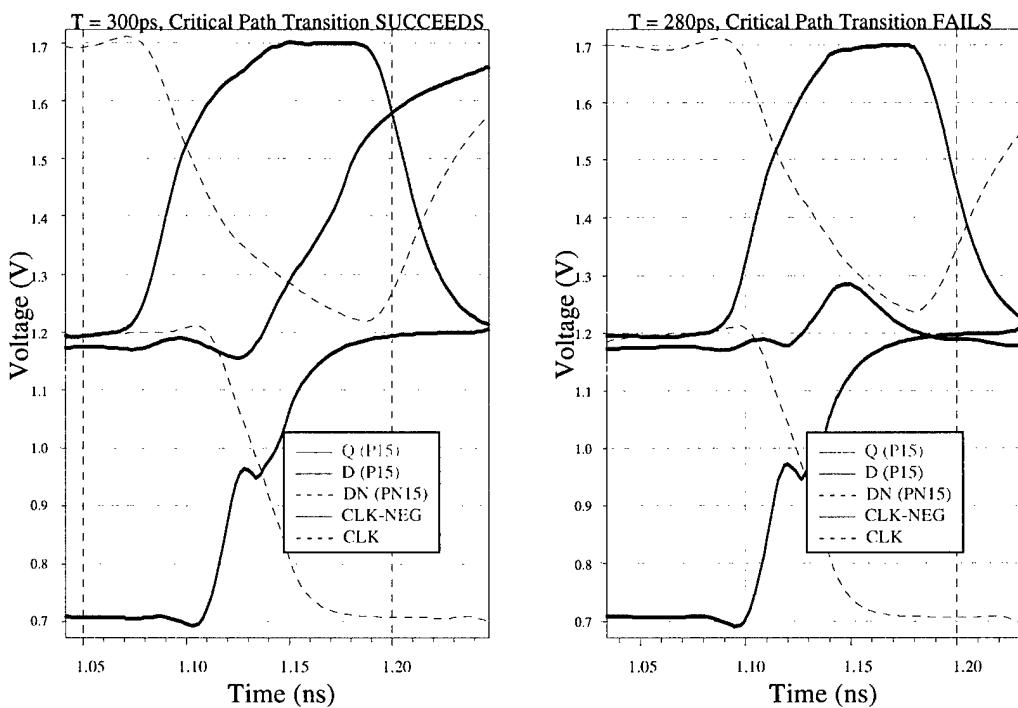
**Figure 5-12.** **Performance bracket of the minimum period for the four-stage pipeline multiplier.**

113

A = 1111 1001   B = 0010 0001

16-Bit Input Register

Carry Save Adder #1 (1)

Carry Save Adder #2 (2)

31-Bit Intermediate Register

Carry Save Adder #2 (3)

Carry Save Adder #2 (4)

31-Bit Intermediate Register

Carry Save Adder #2 (5)

Carry Save Adder #2 (6)

23-Bit Intermediate Register

Carry Completion Adder
(3 Bits)

21-Bit Intermediate Register

Carry Completion Adder
(3 Bits)

Carry

18-Bit Intermediate Register

Carry Completion Adder
(2 Bits)

16-Bit Output Register

Critical Path Initiates
with the two operands
A=F9h, B=21h

Signal Propagation

The Critical Path is Limited
by the LOW-to-HIGH
transition of the Carry Bit
out of Stage 5.

P = 0010 0000 0001 1001

Figure 5-13. Six-stage pipelined multiplier implementation with an illustration of the critical path.

114

	Number of Transistors	Number of Resistors	Current (Amperes)	Power (Watts)
Logic	3952	2352	1.28	3.20
Registers	1872	1560	1.49	3.72
Clock	648	540	1.03	2.57
TOTAL	6472	4452	3.80	9.49

Maximum Throughput: 4.35 GHz

Latency: 1.38 Nano-seconds

Table 5-4.   Performance summary for the six-stage pipelined multiplier.
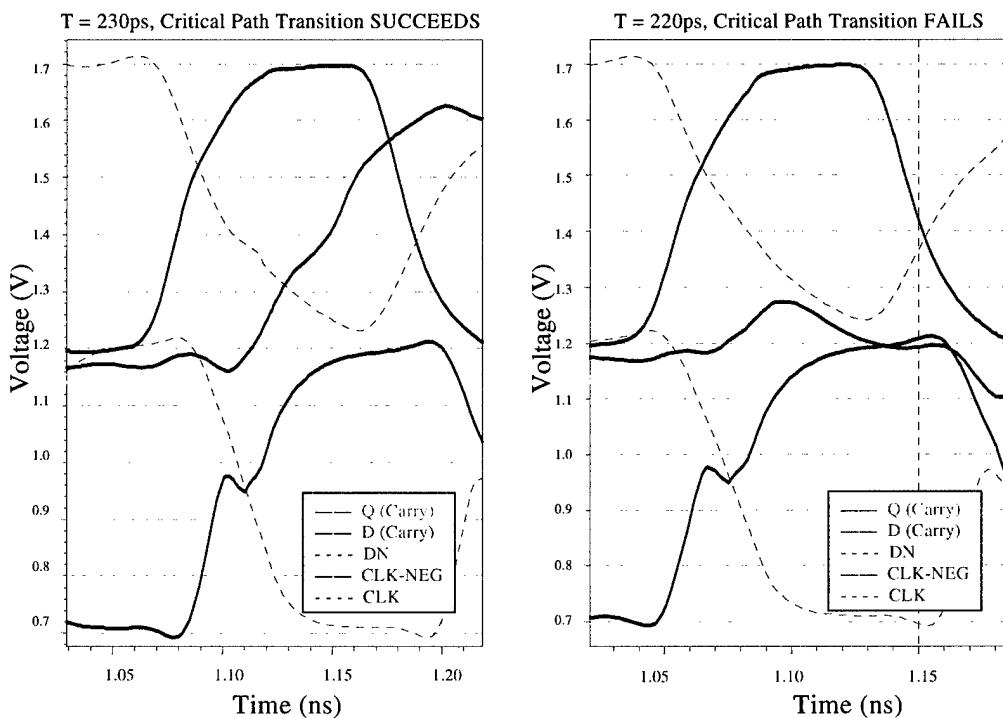

Figure 5-14.   Performance bracket of the minimum period for the six-stage pipeline multiplier.

A = 1111 1001   B = 0010 0001

Critical Path Initiates
with the two operands
A=F9h, B=21h

16-Bit Input Register
Carry Save Adder #1 (1)
31-Bit Intermediate Register
Carry Save Adder #2 (2)
31-Bit Intermediate Register
Carry Save Adder #2 (3)
31-Bit Intermediate Register
Carry Save Adder #2 (4)
31-Bit Intermediate Register
Carry Save Adder #2 (5)
31-Bit Intermediate Register
Carry Save Adder #2 (6)
23-Bit Intermediate Register
Carry Completion Adder (2 Bits)
22-Bit Intermediate Register
Carry Completion Adder (2 Bits)
20-Bit Intermediate Register
C Carry Completion Adder (2 Bits)
18-Bit Intermediate Register
Carry Completion Adder (2 Bits)
16-Bit Output Register

Signal Propagation

The Critical Path is Limited
by the LOW-to-HIGH
transition of the Carry Bit
out of Stage 9.

P = 0010 0000 0001 1001

**Figure 5-15.   Ten-stage pipelined multiplier
implementation with an illustration of the
critical path.**

116

	Number of Transistors	Number of Resistors	Current (Amperes)	Power (Watts)
Logic	3912	2320	1.28	3.20
Registers	3240	2700	2.57	6.44
Clock	1116	930	1.74	4.36
TOTAL	8268	5950	5.60	13.99

Maximum Throughput: 5.56 GHz

Latency: 1.80 Nano-seconds

Table 5-5. Performance summary for the ten-stage pipelined multiplier.
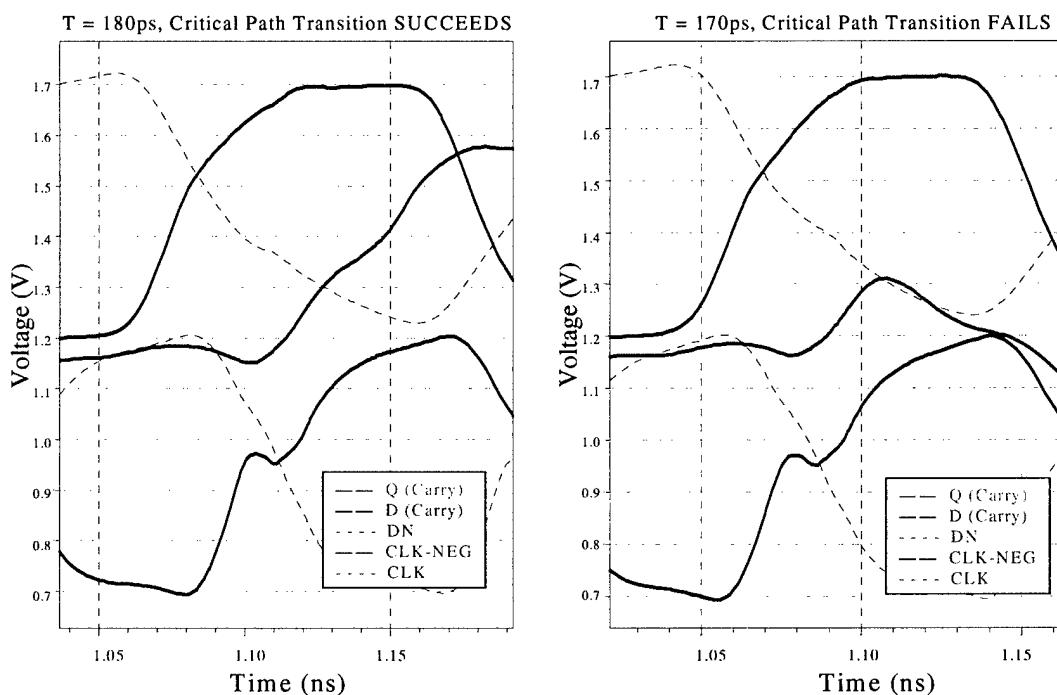

Figure 5-16. Performance bracket of the minimum period for the ten-stage pipeline multiplier.

117

3. Comparative Analysis

A summary of the performance results for each of the five pipelined multiplier implementations is presented in Table (5-6). A comparative analysis of these results quantifies and confirms the major trade-offs of pipelining as they were addressed in Chapter II-B. Figure (5-17) illustrates the increase in data throughput as compared to the increase in product latency. However, latency is generally an acceptable trade-off relative to the primary cost drivers of device count and power consumption.

	1 STAGE	2 STAGE	4 STAGE	6 STAGE	10 STAGE
Device Count	7239	7932	9439	10924	14218
Power (Watts)	4.44	5.41	7.43	9.49	13.99
Latency (nS)	0.75	1.00	1.20	1.38	1.80
Maximum Throughput (GHz)	1.33	2.00	3.33	4.35	5.56
Speed-Power Ratio (GHz/Watt)	0.300	0.370	0.449	0.458	0.397
Normalized Speed-Power Ratio	0.66	0.81	0.98	1.00	0.87

Table 5-6. Comparative Summary of Performance.
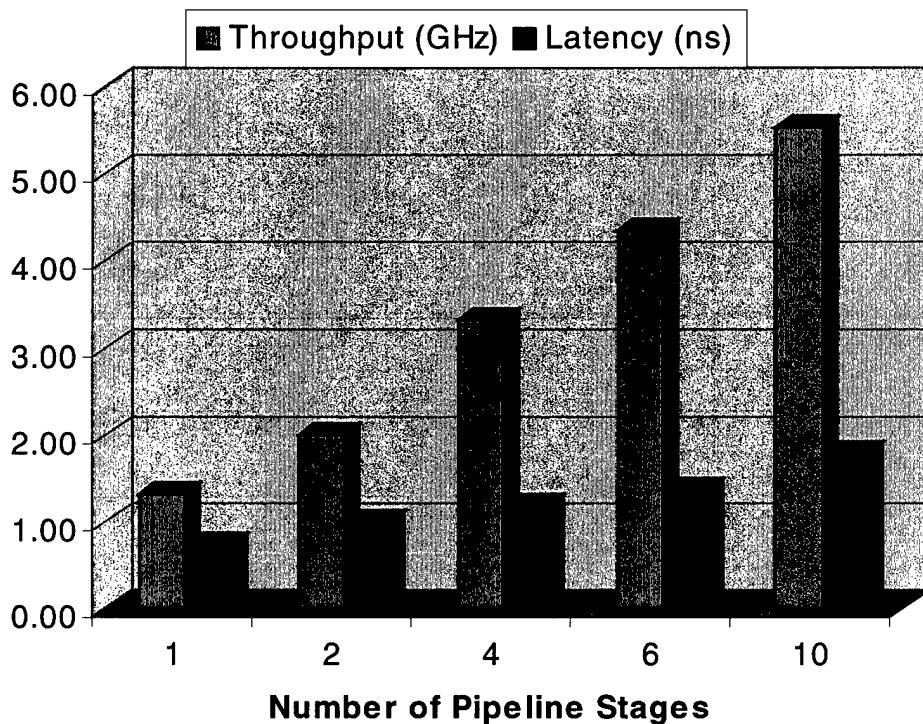
Figure 5-17. Throughput and Latency as a function of the number of pipeline stages.

Device count and power consumption are quantified in Figures (5-18) and (5-19), respectively. As the number of pipeline stages increases, the cost rises sharply - driven by the need for intermediate registers and an extensive clock distribution network. In the one-stage pipeline, the registers and clock tree represent only 13% of the total device count and consume 28% of the total power. On the other end of the spectrum, registers and clock distribution in the ten-stage pipeline represent 56% of the total device count and consume 77% of the total power.

119
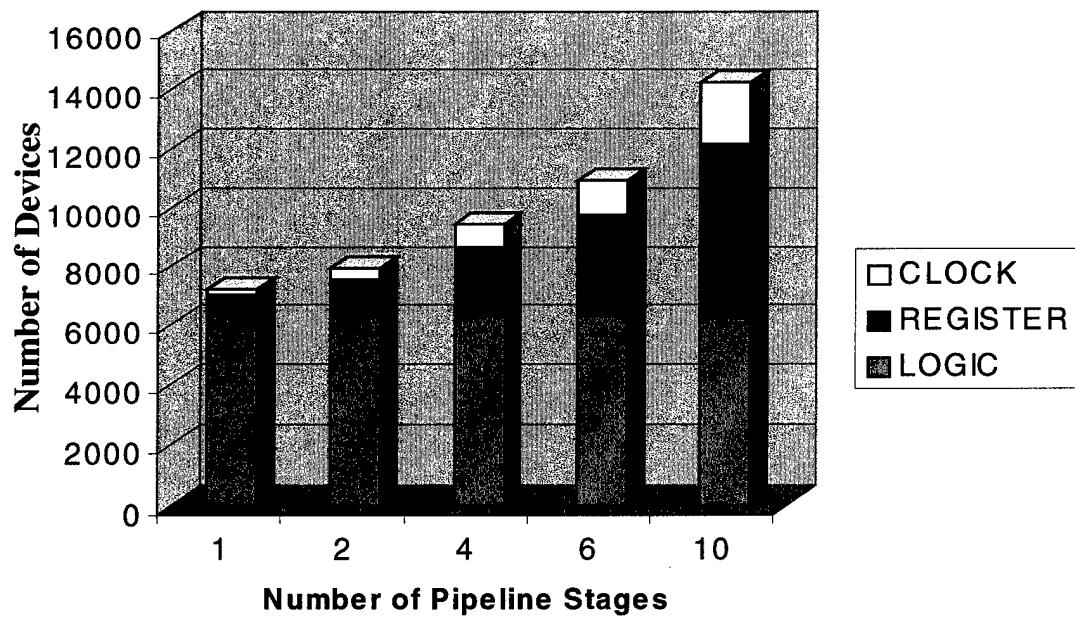
Figure 5-18. Distribution of the Device Count.
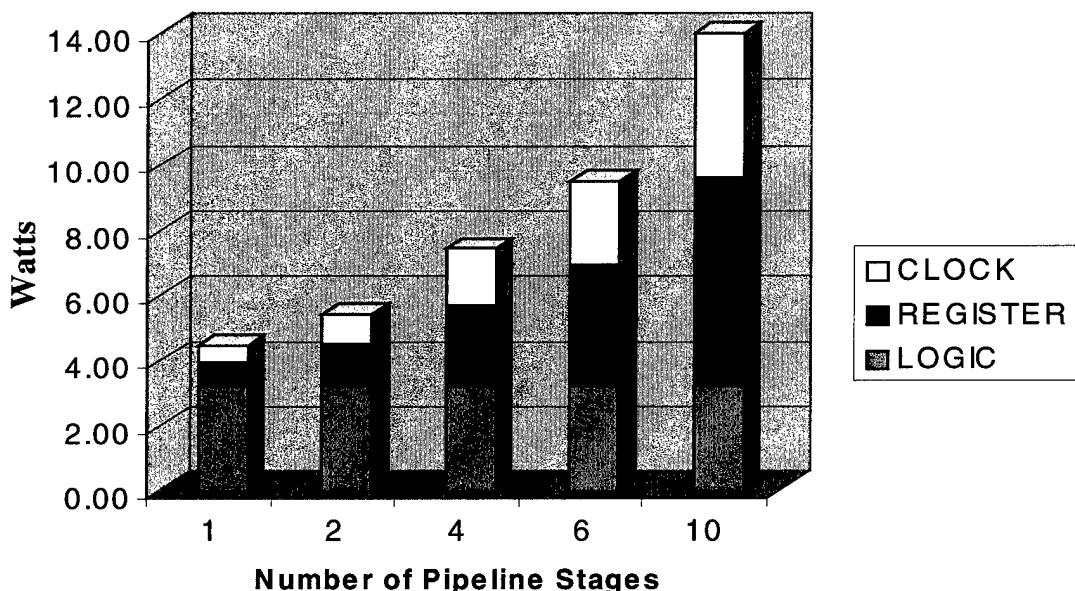

Figure 5-19. Distribution of Power Consumption.

120

Somewhere between these two extremes there exists an optimum pipelined implementation. Dividing the maximum throughput of each configuration by the total power that it consumes, a figure of merit is calculated which is referred to here as a speed-power ratio (for consistency with optimization procedures in previous chapters). Figure (5-21) plots the speed-power ratio as a function of the number of pipeline stages. The maximum point on the curve indicates that the optimal pipelined multiplier implementation employs five or six stages.
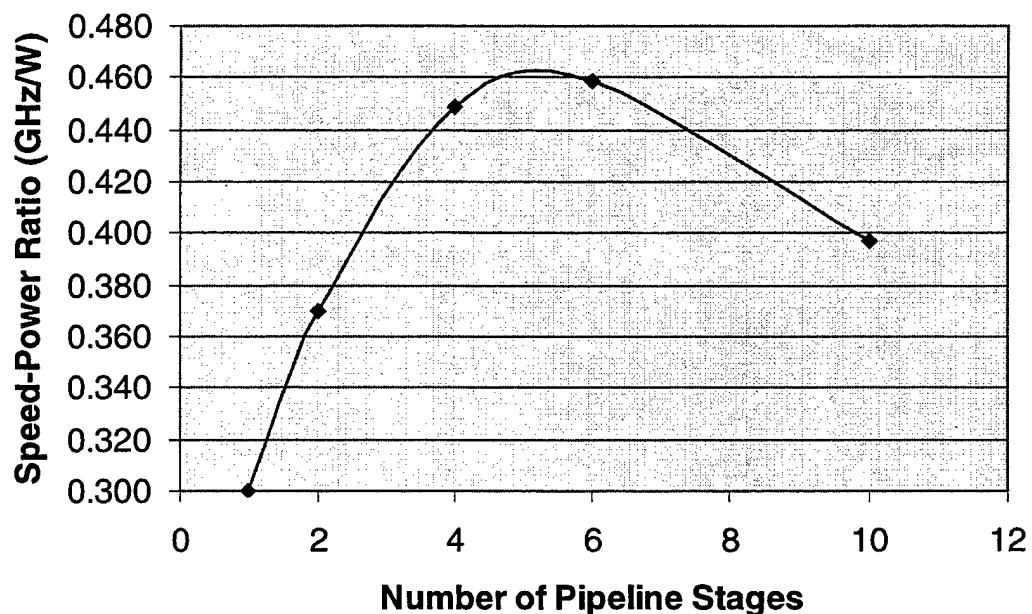

Figure 5-20. Comparison of Speed-Power Ratio.

Thus, having concluded an evaluation of the various pipelined multiplier implementations, it remains to consider the impact that clock skew has upon these high-speed circuits. Chapter VI undertakes this discussion in the pages that follow.

VI.   ANALYSIS OF CLOCK SKEW

A.   QUANTIFYING CLOCK SKEW

Clock skew appears naturally in practical circuits due to a variety of physical factors as described in Chapter II-A. However, in a typical SPICE simulation, transmission delays are not inherent to the process and circuit elements are evaluated under ideal, homogeneous operating conditions. The effective result is the near elimination of clock skew from the simulation environment.

Clock skew could be introduced artificially; however, introducing a known amount of clock skew would have very predictable results, such that it can be determined without simulation. Thus, based upon the results of Chapter V a simple numerical analysis is conducted in this chapter which provides an illustration of how clock skew impacts pipelined architectures and serves as a set of reference data from which follow-on research into alternative control techniques can measure performance.

B.   ANALYSIS PROCEDURES

Based upon the definition of skew from Chapter II-A, let S_{DEVICE} represent the maximum delay between two clock signals after propagation through a single level of clock drivers. As illustrated in Figure (6-1), the effect of S_{DEVICE} on the clock signal as it propagates through the clock

distribution tree is that the clock signal potentially accumulates S_{DRIVER} picoseconds of skew at each level. Furthermore, any loading differences at the final level of the clock distribution will introduce another skew term, S_{LOAD}. Thus, the simplified expression to be used for analyzing and calculating skew is given in Equation (6-1).

(6-1)
$$S_{TOTAL} = n \times S_{DEVICE} + S_{LOAD}$$

$$\text{where, } n = \text{maximum number of levels in the clock distribution scheme}$$
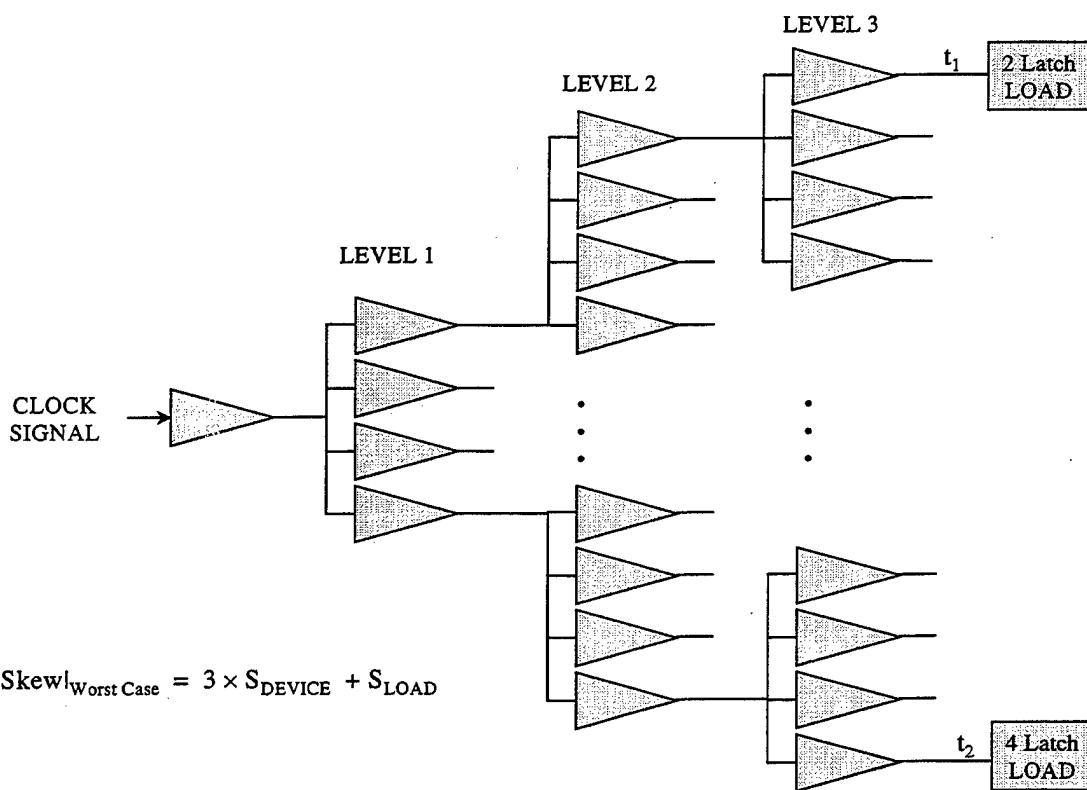

Figure 6-1. Illustration of Clock Skew as it results from propagation path delays and loading.

An expression for *n* is derived in Equation (6-2), based upon the pipeline implementations from Chapter V.

(6-2)
$$n = \left\lceil \log_4\left(\frac{\# \text{REG}}{2}\right) \right\rceil$$

where, $\#\text{REG} = 32 + 26.4(p-1)$

p = Number of Pipeline Levels

For synchronous logic, the timing inequality from Chapter II-A is repeated as Equation (6-3). This relationship requires that the minimum clock period be expanded to account for the increase in skew.

(6-3)
$$T_{min} = t_{skew} + t_{logic} + t_{Flip-Flop}$$

The procedure for analysis of clock skew is simply to apply a range of values for S_{DEVICE} to the clock distribution schemes from Chapter V, using Equation (6-2). Based upon simulation results, the worst-case value for S_{LOAD} is determined to be 6.5 picoseconds. Thus, it is possible to calculate a worst-case skew value for each incremental value of S_{DEVICE} as it applies to the clock distribution scheme of each multiplier implementation. Applying the worst-case skew values to Equation (6-3), a new minimum period is determined for each multiplier implementation. This is

125

repeated for values of S_{DEVICE} ranging from two to twenty picoseconds. A comparative analysis of the results should identify/confirm the expectation of an increasingly negative impact on the more heavily pipelined architectures.

Finally, within the stated range of S_{DEVICE} values, a reasonable figure for S_{DEVICE} is determined as it might actually occur due to device non-idealities in the fabrication process. The approximation of device-induced skew (S_{DEVICE}) is defined as 20% of the worst-case propagation delay for the clock driver circuit and is determined to be 4.5 picoseconds. This set of data is referenced in the figures that follow as "typical skew".

C. RESULTS

Figure (6-2) provides a plot of the results. The values for skew which are referenced in the figures represent the values for S_{DEVICE}. The data clearly confirms that the multipliers with throughput rates which are obtained as a function of higher clock rates will experience the most drastic performance reductions in the presence of clock skew. Furthermore, when weighed against the cost of power consumption a set of new speed-ratio curves is obtained, as shown in Figure (6-3). Thus, the contemporary appeal of synchronous pipelined architectures demonstrates a severe backlash at high clock rates.
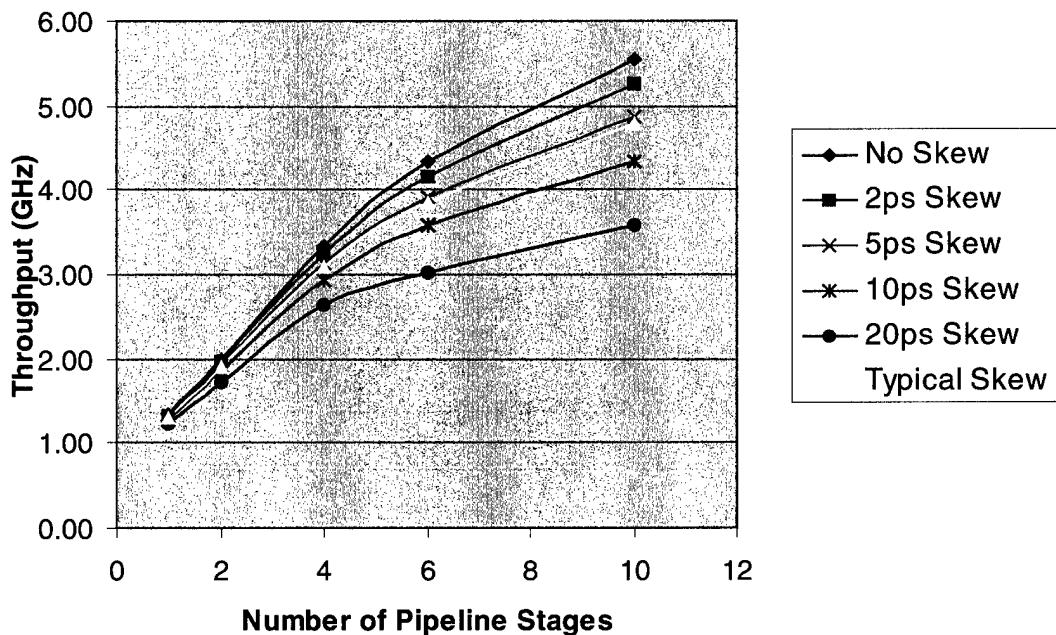
**Figure 6-2. Effect of Skew on Pipeline Throughput Rates.**
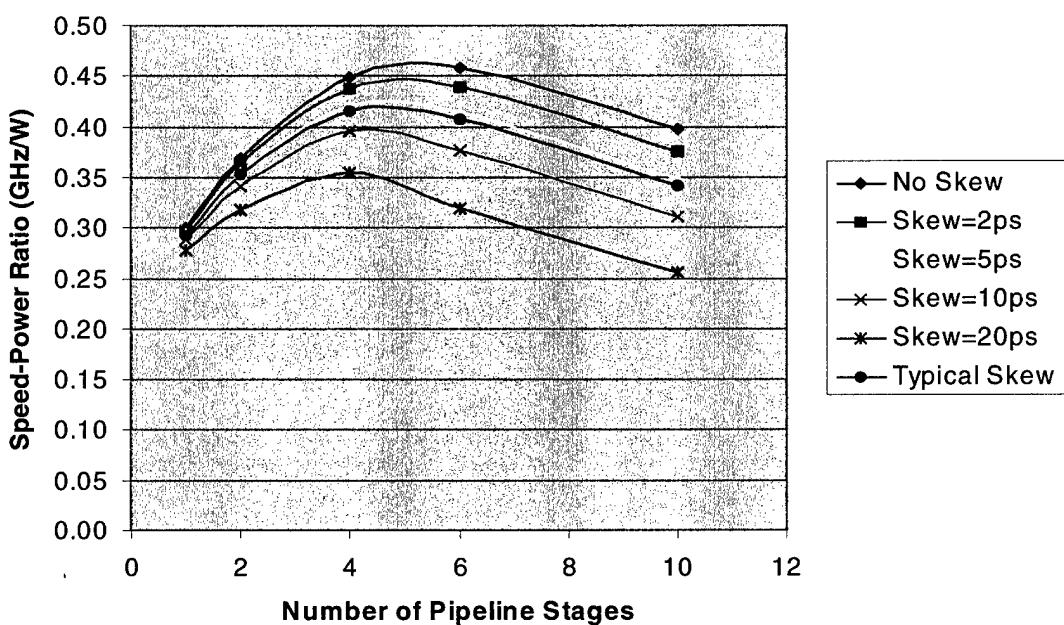
Figure 6-3. Effect of Skew on Pipeline Efficiency.

THIS PAGE LEFT BLANK INTENTIONALLY

VII. CONCLUSIONS

The fundamentals of circuit analysis and the principles of junction transistor behavior have been applied to design an optimal family of current-mode logic devices from InP HBT SPICE transistor models. From these building blocks of digital logic, an array multiplier has been constructed and pipelined into five distinct implementations. Each multiplier implementation has been simulated extensively via Tanner SPICE in order to identify the respective performance characteristics of power consumption and maximum operating frequency.

A comparative analysis of multiplier performance has effectively demonstrated the trade-offs of pipelining with predictable yet interesting results. The cost of increasing throughput by increasing the number of pipeline stages has been quantified in terms of device count and power consumption. By maximizing data throughput at the most efficient cost in terms of power, the optimal 8x8 bit synchronous pipelined multiplier design has been determined to be the six-stage implementation, as shown on page 121.

Finally, in the presence of clock skew, it has been demonstrated that the efficiency of synchronous pipelined architectures operating at high clock rates is significantly reduced. Thus, as device switching frequencies continue to

pave the way to faster logic circuits, the rate of data throughput will be left behind unless the synchronous logic design constraint of clock skew can be overcome. The impact of clock skew has been quantified and summarized such that it provides a reference point for further research into alternative clocking/control techniques.

Specifically, it is intended that future research use the CML HBT logic family designed in this thesis in order to implement the same array multiplier circuit using asynchronous control techniques. One such endeavor is already in progress as LtCol. Kirk Shawhan, USMC, investigates the use of local completion signals which employ request/acknowledge handshake signals to control the flow of data vice the use of a global clock signal (Shawhan, 2000). Perhaps in time such asynchronous schemes will mature into a design methodology that overcomes the obstacle of clock skew which now threatens to limit synchronous design methodology.

LIST OF REFERENCES

Foley, J. B .; Bannister, J. A. R., "Analysing ECL's Noise Margins," *IEEE Circuits and Devices*, Volume 10, 1994, pp. 32-37.

Harris, D., "Timing Analysis Including Clock Skew," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Volume 18, 1999, pp. 1608-1618.

Jalali, B.; Pearton, S. J., *InP HBTs: Growth, Processing, and Applications*, Artech House, Inc., Massachussets, 1995.

Loomis, Herschel H. Jr., "Class Notes," EC 4830 Spring Quarter at the Naval Postgraduate School, 2000.

Moore, G., "Moore's Law Extended: The Return of Cleverness," *Solid State Technology*, Volume 40, 1997, pp. 359-364.

Pierret, Robert F., *Semiconductor Device Fundamentals*, Addison-Wesley Publishing Company, Massachusetts, 1996.

Pollard, L. Howard, *Computer Design and Architecture*, Prentice-Hall, New Jersey, 1990.

Richards, R. K., *Electronic Digital Components and Circuits*, D. Van Nostrand Company, New Jersey, 1967.

Sedra, Adel. S; Smith, Kenneth C., *Microelectronic Circuits*, Oxford University Press, New York, 1998.

Shawhan, Kirk A., *Design and Analysis of an Asynchronous Pipelined Multiplier with Comparison to Synchronous Implementation*, Master Thesis, Naval Postgraduate School, Monterey, CA, Dec 2000.

Sutherland, Ivan E., "Micropipelines," *Communications of the ACM*, Volume 32, 1989, 720-738.

Wakerly, John F., *Digital Design Principles and Practices*, Prentice Hall, New Jersey, 2000.

Weste, Niel H. E.; Eshraghian, Kamran, *Principles of CMOS VLSI Design: A Systems Perspective*, Addison Wesley Longman, Inc., 1993.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
   8725 John J. Kingman Road, Ste 0944
   Fort Belvoir, VA 22060-6218

2. Dudley Knox Library 2
   Naval Postgraduate School
   411 Dyer Road
   Monterey, California 93943-5101

3. Director, Training and Education 1
   MCCDC, Code C46
   1019 Elliot Rd.
   Quantico, Virginia 22134-5027

4. Director, Marine Corps Research Center 2
   MCCDC, Code C40RC
   2040 Broadway Street
   Quantico, Virginia 22134-5107

5. Marine Corps Tactical System Support Activity 1
   Technical Advisory Branch
   Attn: Librarian
   Box 555171
   Camp Pendleton, CA 92055-5080

6. Marine Corps Representative 1
   Naval Postgraduate School
   Code 037, Bldg. 330, Ingersoll Hall, Room 116
   555 Dyer Road
   Monterey, CA 93943

7. Engineering and Technology Curricular Office, Code 34 1
   Naval Postgraduate School
   Monterey, California 93943-5109

8. Chairman, Code EC 1
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California 93943-5121

9. Professor Douglas Fouts, Code EC/FS 1
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California 93943-5121